Legal knowledge based systems
JURIX 93
Intelligent Tools for Drafting Legislation, Computer - Supported Comparison of Law

The Foundation for Legal Knowledge Systems
Editors:
J.S. Svensson
J.G.J. Wassink
B. van Buggenhout

Patries Kordelaar, Supporting the drafting of a new dutch national assistance act with ExpertiSZe, in: J.S. Svensson, J.G.J. Wassink, B. van Buggenhout (eds.), Legal knowledge based systems JURIX 95: Intelligent Tools for Drafting Legislation, Computer - Supported Comparison of Law, Lelystad: Koninklijke Vermande, pp. 95-106, 1993 ISBN 90 5458 089 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

 http://jurix.bsk.utwente.nl/

SUPPORTING THE DRAFTING OF A NEW DUTCH NATIONAL ASSISTANCE ACT WITH EXPERTISZE

Patries Kordelaar

Summary

This article describes how and with what results the knowledge-based system ExpertiSZe has supported the drafting process of a new Dutch National Assistance Act. I shall give some examples of defects we found when we applied ExpertiSZe to a preliminary draft of this act. The article is concluded with a discussion about the value of the support ExpertiSZe has offered.

1. Introduction

In the Dutch social security domain legislation is an important means for regulation and control. The decision to draft new or additional legislation is usually taken when the present legislation no longer produces the intended results or, in other words, when a problem has arisen in respect of the present legislation.
Legislative drafting is a highly creative process [Thornton, 1987, chapter 6] [Voermans, 1990, p.323]. As a result of this the possibilities of automated support for the drafting of legislation seem limited. Various scientific publications however have pointed to the feasability of supporting the drafting process with knowledge-based-system techniques to improve the quality of bills [Allen, 1980, p.75-100] [Bench-Capon, 1987, p.181-189] [Oskamp, 1990, p.74-76] [Voermans, 1990, p.323-336] [Svensson e.a.,1992, p.51]. Up until now relatively little was known about the actual results this kind of support has yielded (some results can be found in [Den Haan, 1992] and [Svensson, 1993]).
From November 1990 until March 1991 the drafting process of the Herinrichting van de Algemene Bijstandswet (the Revision of the Dutch National Assistance Act) was supported by the knowledge-based system ExpertiSZe[1]. This article describes how and with what results ExpertiSZe supported the drafting process of the Revision. Paragraph two gives an description of the method with which the support took place. In paragraph three the results of the support are presented and illustrated with examples. Paragraph four addresses the problem of ascertaining the value of the results.

2. Using ExpertiSZe to support the drafting process

In this paragraph I will first clarify why the preliminary draft of the Revision was chosen as a research object. Next a description is given of the manner in which ExpertiSZe was applied to support the drafting process of this preliminary draft. In paragraph three and four the results of this support will be discussed.

2.1. The choice of the preliminary draft of the Revision as a research object

The knowledge-based system ExpertiSZe supports the legislator in determining and analysing the (juridical and socio-economic) consequences of social security legislation [Nieuwenhuis e.a., 1989] [Svensson e.a., 1992]. ExpertiSZe has a number of modules available for this, of which one, the consistency module, is of particular relevance in this article (for information about the other two modules see [Nieuwenhuis, 1989] and [Svensson, 1993]).
Drafting legislation is a decision-making process in which the drafter, after identifying the problem which has arisen with respect to the present legislation, generates alternative solutions and tests the consequences of those solutions previous to making a choice [Simon, 1976, p.67] [Bench-Capon, 1987, p.184]. While testing the alternative solutions, the determination and analysis of consequences plays an important role.

J.S. Svensson, J.G.J. Wassink and B. van Buggenhout (eds.) (1993). Legal Knowledge Based Systems: Jurix '93: Intelligent Tools for Drafting Legislation, Computer-Supported Comparison of Law.

Since ExpertiSZe supports the determination and analysis of consequences, ExpertiSZe can be usefully deployed during the testing of alternative solutions [Nieuwenhuis e.a., 1989, p.177-178].
Object of research in the project described in this article was the preliminary draft of the Revision of the Dutch National Assistance Act (number 21084) as presented to the Dutch National Economic Development Council (Sociaal-Economische Raad) in August 1990. A preliminary draft can be considered as an alternative solution, so using ExpertiSZe to test such a draft is in theory possible.
Before the preliminary draft of the Revision was presented to the Dutch National Economic Development Council for advice, drafters of the Ministry of Social Affairs and Employment had worked on this draft for two years. During this period several interest groups and other persons and organisations working in the social security domain (such as for example executioners of the National Assistance Act) were frequently consulted. One can legitimately say that the draft was well-considered, advanced and stable. The testing of the Revision draft with the help of ExpertiSZe was expected not to bring up much new information about the consequences of the draft.

2.2. The application of ExpertiSZe

The support ExpertiSZe gives can be divided into two stages: the stage of modelling the legislation and that of the exploration of the model. In the modelling stage a model of the legislation is made, which is processed during the exploration stage.

Modelling the legislation
Using ExpertiSZe a model of social security legislation can be made in a knowledge base. To make this model, the Knowledge Representation Language KRL is used, which was developed in Nieuwenhuis and Svensson's research [Nieuwenhuis, 1989]. Modelling implies that the text of the legislation is represented in KRL. To minimize the chance of errors and misinterpretations in the knowledge base we try to make a representation in KRL which corresponds as closely as possible to the text of the law, where form and content are concerned [Nieuwenhuis, 1989, p.48-49], Svensson e.a., 1992].
KRL is a formal language. Representing legislation in a formal language requires greater precision than representing that legislation in natural language. KRL forces the person who represents legislation in the knowledge base to make the content of that legislation explicit and clear. If it is not possible to represent legislation in KRL, this may be an indication that the legislation is defective on that point[2]. The legislation can for example be incomplete or ambiguous. Defects in the legislation can obstruct the representation of that legislation in the knowledge base. To be able to use the model during the exploration stage the legislation must, after taking notice of the defects, be represented according to the most likely interpretation.
After the representation of the legislation in the knowledge base, the knowledge base is compiled. During that compilation ExpertiSZe checks whether the knowledge base is complete and lacks redundancies and loops. If ExpertiSZe spots lacunae, redundancies or loops in the knowledge base, this can mean that the legislation represented contains defects. Again these defects must be bettered according to the most likely interpretation of the legislation.
In the project described in this article a model of thirteen sections of the preliminary draft of the Revision was made. These sections form a substantial part of this preliminary draft, as far as importance is concerned. With these sections one can determine whether a client is entitled to a benefit and what the amount of the benefit will be. The modelling activity has been carried out by a person who was familiar with the working of ExpertiSZe but who had no appreciable knowledge of the Dutch National Assistance act. This person has been supported by one of the authors of the preliminary draft in order to get the necessary information about the most likely interpretation of the draft.

Exploration of the model
The result of the modelling stage, the model of the legislation, can be processed with the help of the consistency module. Processing means that on the basis of the model the results of the legislation are calculated and tested against requirements which can be placed upon the legislation. An example of such a requirement is that like cases should receive like treatment, and that unlike cases should be treated differentially to the degree of their dissimilarity (the principle of equality before the law). If the results don't comply with a requirement, this can mean that there is a defect in the legislation which was modelled.
The consistency module makes it possible to process the model in the knowledge base for large volumes of theoretical cases. These theoretical cases, which can occur in reality, are generated automatically by ExpertiSZe. This generation takes place on the basis of the definitions in the knowledge base. Generation of cases on the basis of definitions in the knowledge base means the following:
In legislation, distinctions are made between different classes of persons on the basis of certain characteristics. Since the knowledge base contains a representation of that legislation, those distinguishing characteristics are also present in the knowledge base. The consistency module generates all different combinations of distinguishing characteristics which are possible on the basis of the knowledge base. Each combination of characteristics forms a case and each case represents a certain class of persons. Thus if the legislation and the knowledge base contain a rule which applies to a "single person aged 21 or older", then ExpertiSZe automatically constructs the cases "single person aged 21 or older", "non-single person aged 21 or older", "single person younger than 21" and "non-single person younger than 21".
Once all cases that can possibly be derived from the knowledge base have been generated, ExpertiSZe automatically calculates the results for those cases (for example, the level of a benefit payment), according to the knowledge base. These results can then be tested against requirements which can placed upon the legislation.
With the help of the consistency module all cases that are possible on the basis of the model of the Revision were generated and all results for these cases were calculated. The results were tested against requirements which were formulated beforehand by one of the authors of the Revision (an example of such a requirement is: the amount of the benefit must increase as the age of the applicant rises). If the results did not comply with the requirements that same author was asked to give an indication of the source of this conflict.

3. The results

Thirteen sections of the preliminary draft of the Revision were modelled in the knowledge base and explored with the help of the consistency module. As a result **twenty-six defects** in the legislation were found. **Twenty-one defects** were discovered while modelling the legislation and **five defects** were found during the exploration stage.
In this paragraph I present a few examples of the defects found with the help of ExpertiSZe. Also an attempt is made to classify the defects. The fourth paragraph discusses the meaning of the results described in this paragraph.

3.1. Modelling the legislation
The defects found in the preliminary draft of the Revision during the modelling stage all had to do with the legislation not being logically correct or not being clear [Svensson e.a., 1992, p.53]. The legislation contained:
- *9 lacunae:*
  legislation contains lacunae if definitions of concepts and/or (part of) rules are missing;
- *10 ambiguïties:*
  legislation is ambiguous if a part of the legislation can have more than one meaning or interpretation;

- *1 reasoning loop:*
  legislation contains a reasoning loop when the value of a variable in that legislation (for example the amount of income) can only be calculated if that value is already known beforehand;
- *1 indistinctiveness*
  legislation is indistinctive if the meaning of a concept or a rule is vague.

If legislation is not logically correct or lacks clearness it is not possible to come to a (unambiguous) result on the basis of that legislation.
During the compilation no defects were found.

Below the defects found during the modelling stage are illustrated with a few examples[3].

Lacunae

- Example 1

  Section 24(2) states:
  "The total amount of benefit equals the difference between the standard benefit and the income of the single person or the household."

  So: total amount of benefit = standard benefit - income

  Paragraph 2 of the first division of the Revision is titled: "The standard benefit". Paragraph 3 and 4 of that division are titled respectively: "Decrease of the standard benefit" and "Increase of the standard benefit".
  Interpreting section 24(2) literally means that the total amount of benefit can be obtained by subtracting the income from the result of paragraph 2 (the standard benefit). While representing section 24(2) in this literal way we found out that Paragraph 3 and 4 of the Revision remained completely unused. This was not in accordance with the meaning the legislator had given to section 24(2). The legislator wanted section 24(2) be interpreted as follows:

  total amount of benefit = (standard benefit of paragraph 2 - *decrease of paragraph 3 + increase of paragraph 4*) - income.

  Therefore a better formulation of section 24(2) would have been:
  "The total amount of benefit equals the difference between the standard benefit minus the result of paragraph 3 plus the result of paragraph 4 on the one hand, and the income on the other hand.

- *Example 2*

  While representing the legislation in KRL we found that the following complex concepts were not defined:
  - *fellow-occupant* (of a house) in section 34(1);
  - *house* in section 34(1).

Ambiguïties

- *Example 1*

  Section 35(1) states:
  "In as far as the fellow-occupants of the applicant are children younger than 21....."

  In trying to describe this section in KRL it appeared that the phrase "in as far as" could have two different meanings. Must this prescription only be applied when *all* the fellow-occupants are children younger than 21, or can it also be applied when

just *one* of more fellow-occupants is a child younger than 21? The first interpretation was meant by the legislator.

So section 35(1) can be better formulated as:
"Only if all the fellow-occupants of the applicant are children younger than 21....."

- *Example 2*

A part of section 53(1a) goes:
".....in as far as the income of the married couple, ..., is higher than the standard benefit for married couples as mentioned in division 1."[4]

While trying to represent this reference in KRL we found that division 1 contained **two** standard benefits for married couples, one for the case that both partners are entitled to a benefit and one for the case that only one of the partners is entitled to a benefit (the other partner can for example not be entitled to a benefit because he is in prison). It is not clear to which of the two standard benefits section 53(1a) is referring.

Indistinctness

- In section 50(1) the following phrase is used:
"If and as long as the applicant cannot independently provide for his necessary means of existence....."

The meaning of the words "independently provide" is obscure or vague.

Reasoning loop

- While representing sections 32(1) and 53(1a) a reasoning loop was found.

Suppose one of the partners (*partner 1*) of a married couple is entitled to a benefit while the other partner (*partner 2*) is not entitled to a benefit cause he lives abroad. To calculate according to section 32(1) the *amount of benefit partner 1* is entitled to one must subtract from the *standard benefit of partner 1* the *income of partner 1* **and** a *part of the income of partner 2*.

In symbols:
(1) $\text{Benefit}_{\text{partner 1}} = \text{Standard\_benefit}_{\text{partner 1}} - (\text{Income}_{\text{partner 1}} + \text{Income\_which\_should\_be\_included}_{\text{partner 2}})$.

With the help of section 53(1a) we can determine the *part of the income of partner 2 which should be taken into account*. Section 53(1a) says:
"If one of the partners of a married couple is not entitled to a benefit, the income of the partner who is not entitled to a benefit is taken into account:
a. in as far as the total income of the married couple, inclusive of the amount of benefit which should be granted, is higher than the standard benefit for married couples as mentioned in division 1."

In symbols:
(2) $\text{Income\_which\_should\_be\_included}_{\text{partner 2}} = \text{Income}_{\text{partner 1}} + \text{Income}_{\text{partner2}} + \text{Benefit}_{\text{partner 1}} - \text{Standard\_benefit}_{\text{married couples}}$.

So to calculate according to formula (1) the *amount of benefit of partner 1* one must know which part of the *income of partner 2 must be taken into account.* To be able to determine according to formula (2) the *income of partner 2 which should be taken into account* one must know which *amount of benefit partner 1* is getting. Sections 32(1) and 53(1a) contain a reasoning loop.

3.2. Exploration of the model

While exploring the model of the preliminary draft of the Revision with the help of the consistency module **five defects** in the legislation were found. A defect occurs when a result of the legislation, calculated by the consistency module, conflicts with a requirement which was placed upon that legislation. An example of such a requirement is the principle of equality before the law. The requirements to be placed upon the legislation must be ascertained before the exploration takes place. Requirements can for example be obtained by searching literature or by consulting experts on the domain of the legislation which is going to be explored. In the research project described in this article the requirements were formulated by one of the authors of the preliminary draft of the Revision. A few examples out of the many requirements he formulated are:

- the amount of benefit for a married couple must not exceed the sum of the benefits each would have gotten if they were treated as single persons;
- the amount of benefit must increase as age rises from 18 to 27;
- the total income of a family (the total income is the sum of the benefit and the income out of labour) must not depend on which member of that family is earning the income out of labour.

The defects which arise during the exploration from violations of the requirements are difficult to classify since the range of requirements which may be placed upon the legislation is innumerable.
One can for example try to classify the defects according to the source of the requirement (political, juridical, economical, etc.) which is violated. Such a classification has the disadvantage that one requirement can have different sources at the same time. Take for instance the requirement which states that the amount of benefit for a married couple may not exceed the sum of the benefits each of them as a single person would have gotten. This requirement can at the same time have a political source (the Minister has given the requirement as an instruction to the draftsman) and a juridical source (the requirement is a concrete rephrasement of the principle of equality before the law).

One can also look at the defects from an information-processing point of view:
Drafting legislation can be seen as a decision-making process in which a drafter formulates alternative solutions for a problem and determines and analyses the consequences of those solutions previous to making a choice. Human beings, and also drafters of legislation, are imperfect decision makers [Inbar, 1979] [Simon, 1981] [Braam,1988] [Nieuwenhuis, 1989]. One of the most significant causes of this is the limited information-processing capacity of human beings [Inbar, 1979] [Simon, 1981, p.63-98] [Davis and Olson, 1984, p.244-250]. Human beings can for example not generate all possible solutions to a problem and they are also not capable of determining all the consequences of these solutions. Due to this limited information-processing capacity of human beings the result of the drafting process, a bill, is bound to exhibit defects.

ExpertiSZe, and especially the consistency module, was developed to complement a drafter at those points where the human mind, because of his limited information-processing capacities, gets into difficulties.
A computer system like ExpertiSZe can, in comparison with a person, generate a large volume of cases much more easily and in a much shorter time. Moreover this is done more systematically. ExpertiSZe is, unlike a person, in a position to rerun a set of legal rules for different cases uniformly. Especially in a situation where a multitude of rules interact with each other in a complex manner a human being is likely to lose the overview, whereas a computer keeps applying the rules systematically.
Another advantage of using a computer system like ExpertiSZe is that cases are generated without respect of persons and insensitive of certain characteristics of the

cases. When a human being generates cases he is inclined to generate those which are obvious, politically controversial or extreme. He generates cases which lie within the scope of his personal framework. Generating cases with the help of ExpertiSZe, insensitive to certain characteristics of the cases, can therefore lead to the discovery of defects which would have remained unrevealed otherwise.
The defects found with the help of ExpertiSZe during the exploration stage showed that the information-processing abilities of human beings were limited at exactly those points where we a priori expected them to be. Two defects found with the help of ExpertiSZe will be described below. The first defect was discovered because ExpertiSZe generates cases without paying attention to certain characteristics of the cases. The second defect was found because ExpertiSZe has the capacity to generate a very large volume of cases and can also apply rules uniformly to those cases.

Example 1: Generating cases insensitive of the characteristics of those cases.

Requirement placed upon the preliminary draft of the Revision which was violated:
The amount of benefit must increase as the age of the applicant rises from 18 to 27.

One of the general rules underlying the Revision says that the need for income increases as age rises from 18 to 27 (persons under 18 are not entitled to a benefit based on this act). Above 27 a rise in age is not supposed not to add to the need for income. This implies that in the Revision draft the amount of benefit must increase as age rises from 18 to 27.
Generating cases with the help of the consistency module displayed that this requirement was violated in the Revision draft for people living in institutions (for example an institution for mental illness). Table 1 shows the cases which were generated by the consistency module on the basis of a model of section 33 in the knowledge base (section 33 provides rules for people living in institutions). It also shows the standard benefit ExpertiSZe has calculated for those cases. In the table it can be seen that the benefit for people living in institutions does not increase as their age rises from 18 to 27. This means that the requirement placed upon the legislation in this example is violated for these class of cases.
In contrast with people living in institutions the amount of benefit for people who are not living in institutions does increase as their age rises form 18 to 27. So far as this example is concerned one can say that the preliminary draft of the Revision is in conflict with the principle of equality before the law.

Type of household	**age of applicant**	**age of partner 1**	**age of partner 2**	**Standard benefit**
single person	18			337
single person	21			337
single person	23			337
single person	27			337
single parent	18			337
single parent	21			337
single parent	23			337
single parent	27			337
partners living together*		**18	21	562
partners living together		18	23	562
partners living together		27	21	562
partners living together		27	23	562

Table 1: Standard benefit for persons living in institutions.

* both partners living in an institution; ** not all possible combinations of age are mentioned here

The example described here is typically an example of a defect found because ExpertiSZe generates cases without paying respect to the characteristics of these cases. A human being, especially a human being with knowledge of the Dutch National Assistence Act, will not easily discover this defect, since the type of case in which it occurs is not a controversial one. The rule laid down in section 33 of the Revision draft already exists for many years and has never been seriously questioned. The computer generates cases in an insensitive manner and without knowledge of the characteristics of the case and so finds the result mentioned in this example.

Example 2: Generating a large amount of cases and applying rules uniformly

Requirement placed upon the preliminary draft of the Revision which was violated:
The total income of a family (the total income is the sum of the amount of benefit and the income out of labour) must not depend on which member of that family is earning the income out of labour.

Section 35(1) states:

> "In as far as the fellow-occupants of the applicant are children younger than 21 who are not chargeable to the applicant, the amount with which, on the basis of section 34, the standard benefit is lowered is at most the sum of the amounts with which the income out of labour of each of these children is higher than:
> a. if this child is 18 years or younger: 509 guilders
> b. if this child is 19 years: 521 guilders
> c. if this child is 20 years: 633 guilders.

Among the cases generated by the consistency module on the basis of this section were the following two of families with two children under 21 who were not chargeable to the applicant:

- *case 1*
  child 1: 17 years, income out of labour = 0 guilders
  child 2: 20 years, income out of labour = 700 guilders.
- *case 2*
  child 1: 17 years, income out of labour = 300 guilders
  child 2: 20 years, income out of labour = 400 guilders.

The total amount of income out of labour earned by the children is the same for *case 1* and *case 2*, namely 700 guilders. The maximum decrease of the standard benefit however varies in the two cases.

- *case 1*
  maximum decrease = absolute((0-509) + (700-633) = 0 + 68 = 68 guilders
- *case 2*
  maximum decrease = absolute((300-509) + (400-633) = 0 + 0 = 0 guilders.

In a situation like *case 2* there will never be a decrease of the standard benefit on the ground of section 34 because the maximum decrease is zero. In a situation like *case 1* however the standard benefit can be lowered with a maximum of 68 guilders. It can therefore be profitable for a family with two or more children if the income out of labour is earned by more than one child, compared to the situation where one child is earning the income. In other words the total amount of income of a family depends on which member of that family is earning the income out of labour. The requirement discussed in this example is violated. Section 35(1) fosters a situation of unequality before the law.

4. Discussion and conclusions

With the help of ExpertiSZe twenty-six defects were discovered in thirteen sections of the preliminary draft of the Revision. Finding these defects can be seen as the benefits of this project. Opposite to these benefits stand the costs of the project. The finding of the defects has required an input of time and manpower (the project took five months of which three months were used to represent the legislation in the knowledge base). The question is: what is the value of the results of this project or, in other words, how do the benefits of this project relate to the costs?

The value of the results depends on the seriousness of the discovered defects[5] and on whether or not these defects could also have been found without the use of ExpertiSZe. The following remarks can be made with respect to these two points.

Seriousness of the defects

The seriousness of a defect becomes clear during the application of the legislation. It depends in the first place on:

- how often do cases appear in reality in which the defect plays a role;
- how great is the mistake (in terms of costs) which the defect brings about.

Furthermore the seriousness of a defect depends heavily on the chance that the defect will cause an appeal procedure and on the negative influence of the defect on the acceptance of the legislation.

Especially an appeal procedure (including compensation for damages) may carry with it enormous costs. But that is not all. If legislation leads to appeal procedures one must also consider the costs of new amendments that may arise out of those procedures. The chance that a defect will cause an appeal procedure or will eventually make an amendment necessary is difficult to establish.

Defects in legislation can have a negative influence on the social acceptance of that legislation. As a consequence of this, there may be attempts to circumvent the law or attempts to contest it. Defects also have a negative influence on the acceptance of the legislation by the implementing agencies. In the case of defects the danger increases that the law is not applied, or is not applied well. This can mean that the intended results are not achieved. Ultimately this can lead to a new amendment. Again this can bring about enormous costs, but the question whether a defect will actually lead to those costs is almost impossible to answer.

Could the defects also have been found without the use of ExpertiSZe

The question is whether the defects discovered with ExpertiSZe could also have been found without ExpertiSZe. The discovered defects are of a nature that they could just as easily have been found by a human being alone. Nevertheless for a period of two years legal experts of a ministry have worked on the draft and during that period independent organisations (such as organisations which enforce the law) have regularly been studying on the draft. Still they did not detect the defects which were discovered with the help of ExpertiSZe.

Obviously it is difficult, if not impossible, to make a law without defects. One can compare this with searching a text for spelling-mistakes. Despite thorough examination a few mistakes are bound to stay behind. Moreover a human being cannot rerun a set of legal rules for different cases uniformly without consciously or unconsciously interpreting the rules and changing them in order to compensate for defects in the legislation. The reasoning loop in the Revision draft, discussed at page 6, is an example of this situation. One of the authors of the draft compensated for this loop by applying an additional rule which was not mentioned in the legislation, but existed only in his mind. He was not aware of the fact that he was doing this.

The fact that for two years legal experts have studied on the Revision draft and did not discover the defects found with the help of ExpertiSZe indicates that the testing of legislation with ExpertiSZe has additional value over and above manual testing of legislation. The exact value offered by ExpertiSZe can only be ascertained on the basis of a comparative, quasi experimental investigation of the testing of legislation with and without ExpertiSZe. Such an experiment is in preparation.

Conclusion
The results of this research project clearly show that it is possible to support the drafting process with ExpertiSZe. In a stable, advanced and well-considered draft twenty-six defects were found with the help of ExpertiSZe. These defects are bound to stay behind because of the limited information-processing capacities of human beings. A computer system like ExpertiSZe complements a drafter at exactly those points where the human mind, because of its limited information-processing capacities, gets into difficulties.
The defects brought up with the help of ExpertiSZe were found in a draft which has been thoroughly examined by experts for two years. This indicates that testing legislation with ExpertiSZe offers additional value over and above testing legislation without ExpertiSZe. In order to make a definite assessment of the additional value offered by ExpertiSZe a comparative experiment must be designed and executed. Such an experiment is in preparation at this moment.
The defects found during this project in the preliminary draft of the Revision have almost all been reason to alter the text of the draft.

Acknowledgements
The project described in this article forms part of a joint research project of the Ministry of Social Affairs and Employment and the University of Twente. I am greatly indebted to Ton Janush and Gerrit-Jan van 't Eind, both working at the Ministry mentioned in the previous sentence.
Also I thank my promotor Geert Braam and my colleagues at the University for their support. Mrs. Stamper is thanked for translating parts of the article.

Notes
1 ExpertiSZe was developed by the University of Twente during the period 1988-1990. This development took place under the auspices of the Ministry of Social Affairs and Employment.
2 A problem of representation can also be caused by a defect in KRL. In the latter case, KRL will have to be altered. From earlier research [Nieuwenhuis, 1989] [Kordelaar, 1990], it may however be concluded that KRL is suitable for the representation of knowledge with the structure *IF condition THEN action*, and that a great deal of social security legislation can be translated into this form. On the basis of this we expect that representation problems concerning social security legislation may mostly be ascribed to defects in the legislation, and not to the inadequacy of KRL. To be completely sure an expert on the domain of the legislation which is represented (for example the author of the legislation) is asked to give an answer about the source of the defect.
3 It would take far to much space to describe all defects found during the modelling stage. For a full description of the defects I refer to: Kordelaar, P.J.M. and J.G.J. Wassink, *"Bijlagen 1e fase implementatie-onderzoek ExpertiSZe: Ondersteuning Herinrichting Bijstand"*, 1991.
4 The Dutch social security system treats two unmarried people who have the same kind of relationship as married people, as if they were married.
5 What if the modelling and exploring of legislation with the help of ExpertiSZe does not lead to the finding of any defects? Does this mean that there are no benefits at all? We don't think so. If legislation has been tested with the help of ExpertiSZe and no defects have been found, one can at least be more certain that the kind of defects ExpertiSZe is testing on don't occur in the legislation.

References
[Allen, 1980] Allen L.E., Language, law and logic: plain legal drafting for the electronic age, in Niblett, B. (ed.), *Computer Science and Law,* Cambridge: Cambridge University Press, 1980, p.75-100.
[Bench-Capon, 1987] Bench-Capon, T.J.M., Support for policy makers: formulating legislation with the aid of logical models, in: *Proceedings of the First*

International Conference on Artificial Intelligence and Law, Boston: ACM Press, 1987.

[Bench-Capon, 1991] Bench-Capon, T.J.M., Exploiting Isomorphism: development of a KBS to support British coal insurance claims, in: *Proceedings of the Third International Conference on Artificial Intelligence and Law,* Oxford: ACM Press, 1991.

[Braam, 1988] Braam, G.P.A., *Bestuurssociologie,* Alphen aan den Rijn/Brussel: Samson, 1988.

[Davis & Olson, 1984] Davis, G.B. and M.H. Olson, *Management Information Systems: conceptual foundations, structure and development,* New York: McGraw-Hill, 1984.

[Den Haan, 1992] Haan, N. den, TRACS: A support tool for drafting and testing law, in: Grütters, C.A.F.M. e.a. (ed.), *Legal Knowledge-Based Systems: Information Technology & Law, JURIX'92,* Lelystad, Koninklijke Vermande, 1992.

[Inbar, 1979] Inbar, M, *Routine decision making: the future of democracy,* Beverly Hills: Sage, 1979.

[Kordelaar, 1990] Kordelaar, P.J.M., *KRL, een representatietaal voor wetten?,* Enschede: Universiteit Twente, 1990.

[Kordelaar & Wassink, 1991] Kordelaar, P.J.M. en J.G.J. Wassink, *1e fase implementatie-onderzoek ExpertiSZe: Ondersteuning Herinrichting Bijstand,* in opdracht van: Ministerie van Sociale Zaken en Werkgelegenheid, Enschede: Universiteit Twente, 1991.

[Nieuwenhuis, 1989] Nieuwenhuis, M.A., *TESSEC: een expertsysteem voor de Algemene Bijstandswet,* Deventer: Kluwer, 1989 (doctoral dissertation).

[Nieuwenhuis et al., 1989] Nieuwenhuis, M.A., van 't Eind, G.J., Einerhand, M.G.K. en J.G.J. Wassink, ExpertiSZe: een expertsysteem voor de beleidsvoorbereiding, in: van den Herik, H.J. (red.), *AI-Toepassingen '89,* Bussum: De Cirkel, 1989.

[Oskamp, 1990] Oskamp, A., *Het ontwikkelen van juridische expertsystemen,* Antwerpen-Deventer: Kluwer, 1990 (doctoral dissertation).

[Simon, 1976] Simon, H.A., *Administrative Behavior,* New York: The Free Press, 1976.

[Simon, 1981] Simon, H.A., *The Sciences of the Artificial,* Cambridge (Massachusetts): MIT Press, 1981.

[Svensson et al., 1992] Svensson, J.S., Kordelaar, P.J.M., Wassink, J.G.J. en G.J. van 't Eind, ExpertiSZe, a tool for determining the effects of social security legislation, in: Grütters, C.A.F.M. e.a. (ed.), *Legal Knowledge-Based Systems: Information Technology & Law, JURIX'92,* Lelystad, Koninklijke Vermande, 1992.

[Svensson, 1993] Svensson, J.S., *Kennisgebaseerde microsimulatie,* Enschede: Universiteit Twente, 1993 (doctoral dissertation).

[Thornton, 1987] Thornton, G.C., *Legislative drafting,* London: Butterworths, 1987.

[Voermans, 1990] Voermans, W.J.M., Het teken, dat u bewaart en duidt tot in verste verandering, in Backx, H.A.M. (red.), *Recht doen door wetgeving: opstellen over wetgevingsvraagstukken aan Mr. E.M.H. Hirsch Ballin ,* Zwolle: Tjeenk Willink, 1990.