



25th IVR World Congress
LAW SCIENCE AND TECHNOLOGY
Frankfurt am Main
15–20 August 2011

Paper Series

No. 114 / 2012

Series A

Methodology, Logics, Hermeneutics, Linguistics, Law and Finance

Antoinette J. Muntjewerff
Judicial Review in the Democratic
System

URN: urn:nbn:de:hebis:30:3-249726

This paper series has been produced using texts submitted by authors until April 2012.
No responsibility is assumed for the content of abstracts.

Conference Organizers:

Professor Dr. Dr. h.c. Ulfrid Neumann,
Goethe University, Frankfurt/Main
Professor Dr. Klaus Günther, Goethe
University, Frankfurt/Main; Speaker of
the Cluster of Excellence “The Formation
of Normative Orders”
Professor Dr. Lorenz Schulz M.A., Goethe
University, Frankfurt/Main

Edited by:

Goethe University Frankfurt am Main
Department of Law
Grüneburgplatz 1
60629 Frankfurt am Main
Tel.: [+49] (0)69 - 798 34341
Fax: [+49] (0)69 - 798 34523

An Explicit Model for Learning to Structure and Analyze Decisions by Judges

Abstract: Legal practitioners and legal scientists need to have knowledge of the general rules that apply in the legal system. This involves both knowledge of the legislation and knowledge of the decisions by judges that function as general rules of law. Law students preparing themselves for the legal profession need to acquire these kinds of knowledge. A student has to have knowledge about where to look for decisions, understand the structure of decisions and learn to determine what makes a decision relevant to the body of applicable rules in the legal system. Legal education primarily aims at acquiring insight in the legal sources, their history and background. This basic knowledge is of great importance; legal problem solving is hardly possible without an understanding of the legal knowledge. To illustrate the use of this knowledge in practice, teachers work through decisions as examples. However, it is difficult, if not impossible, to learn by explanation or by imitation alone. A more effective way to obtain expertise is by actually performing the task, i.e. students should do the exercises, while the teacher provides feedback on their solutions. For effective learning, also the solution process should be monitored and provided with feedback. Furthermore it is desirable for students to be able to ask for help at any time during the process. They should also be able to practice over and over again. An ideal situation would have a teacher available for every student, monitoring the student while practicing and providing support where and whenever necessary. However, this being not practically feasible, the second best option is to offer the student electronic support.

CASE (Case Analysis and Structuring Environment) is an environment where a law student can practice with finding decisions, with structuring its text and with analysing the decision in order to be able to determine in what way it adds to the body of applicable rules in the legal system.

CASE is developed using a principled and structured design approach. A short description of this approach is followed by an analysis of the learning task, the difficulties law students experience and the remedies proposed on the basis of both the task analysis and the stated difficulties. This is followed by a description of architecture, functionality, platform and implementation of CASE and a description of a session with CASE and future work.

Keywords: Instructional Design, Coaching Systems, Legal Problem Solving

I. Introduction

The law that applies in a legal system such as the Dutch legal system consists of general rules that are determined or acknowledged by authoritative bodies. The two most important authoritative bodies within the Dutch legal system are the legislator and the judge. While it is

obvious the legislator determines rules that apply in general, this is more complicated with judges. A judge has to decide in individual cases, she has to construct a legal solution based on the facts of the case and the applicable legal rules. In the majority of cases that come before the court, a judge formulates a decision that applies only to the case at hand. These decisions do not add to the body of applicable rules in the legal system. However, in cases where a judge first has to construct an applicable rule, before being able to decide the case on the basis of this rule, we have a different type of decision.

The rule constructed by the judge to decide the case, may add to the body of applicable rules in the legal system. Legal practitioners and legal scientists need to have knowledge of the general rules that apply in the legal system. This involves both knowledge of the legislation and knowledge of the decisions by judges that function as general rules of law. Law students preparing themselves for the legal profession also need to acquire knowledge about the role of decisions by judges in the legal system, and they need to understand the two categories of decisions by judges. A student has to have knowledge about where to look for decisions of the second category, understand the structure of decisions and learn to determine what makes a decision relevant to the body of applicable rules in the legal system. Legal education primarily aims at acquiring insight in the legal sources, their history and background. This basic knowledge is of great importance; legal problem solving is hardly possible without an understanding of the legal knowledge. To illustrate the use of this knowledge in practice, teachers work through decisions as examples. However, it is difficult, if not impossible, to learn by explanation or by imitation alone. A more effective way to obtain expertise (skill) is by actually performing the task, i.e. students should do the exercises, while the teacher provides feedback on their solutions. Not only feedback on the solution provided by students is important.

For effective learning, also the solution process should be monitored and provided with feedback. Furthermore it is desirable for students to be able to ask for help at any time during the process. They should also be able to practice over and over again. An ideal situation would have a teacher available for every student, monitoring the student while practicing and providing support where and whenever necessary. However, this being not practically feasible, the second best option is to offer the student electronic support. Using a computer program as the instructional medium has a number of advantages. It may offer individualized instruction and practice combined with immediate support and feedback. It can have the capacity to adapt to the individual student's performance and, last but not least, may support the management of information.

CASE (Case Analysis and Structuring Environment) is an environment where a law student can practice with finding decisions, with structuring its text and with analysing the decision in order to be able to determine in what way it adds to the body of applicable rules in the legal system.

These functionalities are implemented in two integrated modules in CASE:

1. a module to compile and store decisions

In essence a database containing a selection of decisions used in legal education. The law student can do a search (key word and/or full text) for a specific decision or a set of decisions. Decisions can be added to the database and key words can be indicated for each decision by the teacher. This module can be used separately or in combination with the second module.

2. a module to structure and analyse decisions

In essence an instructional environment for learning to structure and analyze a decision to determine how it adds to the body of applicable rules in the legal system. This module builds on the first module. It presents the student the text of a selected decision together with a framework containing the main elements in a decision text (as, for instance, the different parties and their roles in the various stages of their procedures before the different courts). It allows the student to fill the framework with the relevant parts from the text of the decision. The activities of the student are monitored and compared to a model where deviations are diagnosed to be able to present the student with a hint or a remediation.

CASE is developed using the principled and structured design approach as described in the HYPATIA project (Muntjewerff 2002a, Muntjewerff 2002b). A short description of this approach is followed by an analysis of the learning task, the difficulties law students experience and the remedies proposed on the basis of both the task analysis and the stated difficulties. This is followed by a description of architecture, functionality, platform and implementation of CASE and a description of a session with CASE and future work.

II. Principled and Structured Design Approach

The HYPATIA project (Muntjewerff 2002a, Muntjewerff 2002b) aims at designing and developing *new additional* electronic materials for law students to learn the law. Law students experience difficulties in acquiring legal knowledge and in using this knowledge. These problems are acknowledged by law teachers. However, there is no material available to help students to overcome these difficulties. HYPATIA aims to fill this gap developing electronic to offer individualized instruction and practice by adapting to the individual student's performance combined with immediate support and feedback. Electronic tools may also

support the management of information and present different representations and visualizations of legal knowledge and legal tasks. The principled and structured design approach guides the development process in such a way that difficulties and mistakes encountered during the design process may be accounted for. The design process involves two interrelated research streams: basic research and applied research.

Basic research is concerned with developing well-founded models of legal knowledge and skills to be learned by law students, examining the difficulties they have with acquiring legal knowledge and legal skills and finding remedies to enhance effective and efficient learning of the required knowledge and skills. In the applied research part, basic research findings are used to construct computer supported models of legal knowledge and legal reasoning to diagnose and remedy the specific difficulties of law students in learning the law. Instructional design decisions are made on the basis of a global theory on learning and instruction. In this way the design process will result in a coherent and consistent instructional model. It finally indicates that electronic materials are evaluated extensively (developmental testing and field testing). The design approach was founded and used successfully in the construction of an instructional environment for learning to solve legal cases: PROSA.

The approach taken in PROSA is reusable for a variety of applications for learning the law. The legal case solving research within HYPATIA has been realized and reported in detail (see, for instance, Muntjewerff 2000, Muntjewerff and Groothuismink 1999, Muntjewerff 2002c).

III. Analysis

What is structuring and analyzing a decision? In order to answer this question and to design an environment to support law students in finding, reading, structuring and analyzing decisions to indicate and understand the legal meaning of a decision, it is necessary to analyze the task.

The HYPATIA design approach starts with (re)constructing explicit models of legal knowledge and legal reasoning. In this (re) construction process, two components are distinguished. (1) A theoretical component of exploration, conceptualization and specification of legal knowledge and legal reasoning resulting in explicit models of legal knowledge and legal reasoning. Two perspectives are taken within this approach: a legal perspective and a knowledge engineering perspective.

From the legal perspective different legal sources, being legal empirical research, legal educational practice, legal dogmatic and legal theoretical research, are examined to specify

the required models. The knowledge engineering perspective is used to construct models at a high level of explicitness as they have to be executed by a computer (see, for example, Haanden and Sartor 1999, Breuker and Van de Velde 1994, Valente 1995). This explicitness of models is exactly what is needed in instruction. (2) An empirical component where empirical studies are carried out to acquire insight in the way legal practitioners and legal scientists handle legal knowledge in general and in carrying out specific legal tasks. In this case, law students are studied to see how they handle and use legal knowledge to perform a specific legal task and what difficulties they experience.

The legal sources that were examined to model the task of reading and comprehending decisions all describe a series of steps to be taken by the student when reading a decision to determine the legal significance (Bos 2003, Eemeren van et.al. 1996, Franken et.al. 1991, Henket and Hoven van den 1999, Jansen 1999, Scholten 1974, Soeteman and Wolthuis 2003, Stolwijk and Bosch 2002).

However, merely *instructing* a method does not work for novices (see for details Muntjewerff 2000).

This is partly due to the fact that instructing a method is a problem in itself, as it is difficult to communicate a method, because this requires the translation of actions into words.

A method is in fact empty; explaining content is much more “substantial” and therefore easier. The somewhat paradoxical situation is that novices have to learn to determine the legal meaning by determining the legal meaning. Law students especially have difficulties with determining what the decision adds to the body of applicable rules in the legal system.

Based on findings in research in legal problem solving it is stated that the difficulties are first of all caused by insufficient mastery of, or insight in, the subject matter. Secondly, especially for novices, methods, often as a side effect, emerge from (novice) problem solving, instead of being the driving force. The subject matter appears to be the major source for finding or trying (a) solution (steps). On closer inspection, a decision is a legal solution for a specific problem situation constructed on the basis of abstract legal rules. Structuring and analyzing a decision is in fact the task of reconstructing the problem situation (consisting of a reconstruction of both the facts and the legal question), tracing the abstract legal rules that were applied and specifying the legal solution consisting of the argument structure and the conclusion.

Reading and understanding a decision is not a trivial activity. Observations with first year law students reading decisions showed that they experience difficulties with seeing through

the composition of the decision, with reconstructing the argument structure and with determining the legal significance of the decision.

These difficulties are first of all caused by the fact that a decision is an incomplete reproduction of what happened. Next to that the text of the decision contains many references, both explicit and implicit, to regulations, other decisions and concepts. The fact that a decision has a stratified structure which is also not supported by recognizable clues or elements in the text does not help either.

All of this means that the student has to reconstruct the process and the product which involves keeping track of intermediate results. To support the student in performing these tasks, the following remedies are proposed. Present the student a structure to help her to reconstruct the decision, support the management of information and engage the student in structuring and analyzing the decision by having her actually carry out these tasks. This is realized by presenting the student with both the full text of the decision and a framework which visualizes the elements in a decision necessary to reconstruct the decision in order to determine the legal significance of the decision.

There are no applications available that support law students in structuring and analyzing a decision suiting the Continental legal system. For the Anglo-American legal system, the CATO application is available (Aleven 1997). In CATO the student is trained to construct arguments with cases.

IV. Architecture and Modular Design

The aim of the CASE project is to realize an environment in which law students are supported in structuring and analyzing a decision. This means that both the decision at hand has to be presented to the student, as well as the framework for analysis. The student must be able to select text fragments from the decision and paste these within the correct cell in the relevant table in the framework. Since finding cases is also part of the training of law students search facilities have to be available in the environment. The functionality of searching for a decision is implemented in the first module. The functionality of structuring and analyzing a decision is implemented in the second module. Other basic requirements are maintenance and re-use. It should be possible to make changes to the system and its content without much costs and efforts. Errors in system and content should be easily traceable and correctable. It must be possible to add and delete content without causing problems elsewhere in the system. Transparency of the architecture and tools are therefore design goals, as it may facilitate maintenance.

The system has functions for adding decisions, adding key words to decisions and preparing decisions for analysis. System functionalities are attributed to a user on the basis of her status: administrator, editor, teacher or student. The database module holds the decisions and allows for search and retrieval of cases and allows teachers to prepare cases for use in the analysis module. Students can use the database module to locate cases on the basis of key words and/or full text search to find specific decisions. When the student wants to structure and analyze a decision she can select one of the reported decisions. This decision and the analyzing framework are then made available to the student. The student can start structuring the decision by selecting text fragments in the decision and pasting these in the correct part of the frame.

V. Platform and Implementation

CASE is implemented using a web-based server-side application model. The user interacts with the system using a standard web browser, such as Netscape Navigator, Apple Safari or MS Internet Explorer. CASE is developed using Open Source Software, MySQL (4.0.14) and PHP (4.3.2) and JavaScript. The MySQL database backend contains a number of tables, the most prominent ones being a text fragment table, a solution table and a table storing the student's activities. CASE's primary component is the server-side application implemented in PHP (4.3.2). This application handles form processing, storage and retrieval of information from the various tables in the database and generating the HTML pages that are output to the user.

A small number of simple functions are implemented using client-side JavaScripts. CASE offers extensive support for administrative-, editing-, browsing-, tracking- and educational tasks. Using the same portal, administrators can add, remove and change users and cases; editors can add keywords to cases and prepare the solution framework of a case for use; teachers can use the interface to track the results of students, previewing the solution framework and for browsing and searching the database; and students can browse and search the database, and test their analysis skills.

The search engine allows for both Boolean keyword- and free text search in combination with metadata fields such as: date, name, court etc. The principal concept in CASE is that a precedent can be seen as an ordered set of text fragments, each of which can be labeled according to their place in the solution template. The student can select a text fragment and place in a specific position within the solution framework. Text fragments can be as short as a single sentence, but more often, they are as long as a paragraph. The text fragments are stored

in a database along with metadata such as a reference to their position in the solution. Although a text fragment as described is the basic building block, these fragments can have one or more sub-fragments (such as single words) which can also be selected by the student. For instance, the text fragment

“Op het beroep van Ronald G, geboren te Amsterdam op 6 aug. 1954, wonende te Amsterdam, req. van cassatie tegen een bij verstek gewezen arrest van het Hof te Amsterdam van 12 dec. 1977, waarbij in hoger beroep een vonnis van de Rb.”,

contains the sub-fragment 'Ronald G', the accused. In some cases the student needs to select the whole sentence, and in others only the sub fragment. The solution framework consists of a number of tables, such as parties, facts, claim and the argument structure before the Supreme Court. Each table is two dimensional and contains a small number of cells, e.g. facts as presented by the initiator, and facts presented by the opponent.

Each cell in the solution, therefore, can be designated by three coordinates: table, row and column. These coordinates are used to mark the proper location of text fragments within the solution framework.

They allow the student's solution to be tested against the solution defined by the teacher; the cell in which the student places the fragment has to match the metadata reference of the text fragment. In the case of an incorrect placement of a fragment, its position relative to the correct place is also known. This allows for standardized responses to common errors. For instance, when a student puts the initiator's name in the opponent's cell, the following response can be generated on the basis of this mixing up of the parties in the dispute: "This indeed is one of the parties in the dispute, but unfortunately it is not the opponent.". To get a basic idea of the functionality of the system we now describe a session with CASE.

VI. A session with CASE

As mentioned above, CASE distinguishes four types of user: *administrators*, *editors*, *teachers* and *students*. User rights are distributed in an *incremental* fashion in CASE, this means that a teacher has access to both student- and teaching facilities; an editor has access to editing-, teaching- and student facilities; and the administrator user has rights to do everything the other users can, plus adding, removing and changing users, and removing cases from the database. This section describes a typical process from preparation to analysis of a case.

After login, the editor is presented with a menu containing multiple options. Since the editor recently came upon a decision relevant for law students, she decides to add it to the CASE database. The *editor's menu* gives access to the *add decision* screen.

Here she fills in a few facts about the decision (name, publication date, court etc.) and with copy- paste actions she adds the text of the decision to the database. Next, she visits the *metadata editor* (see figure 1).

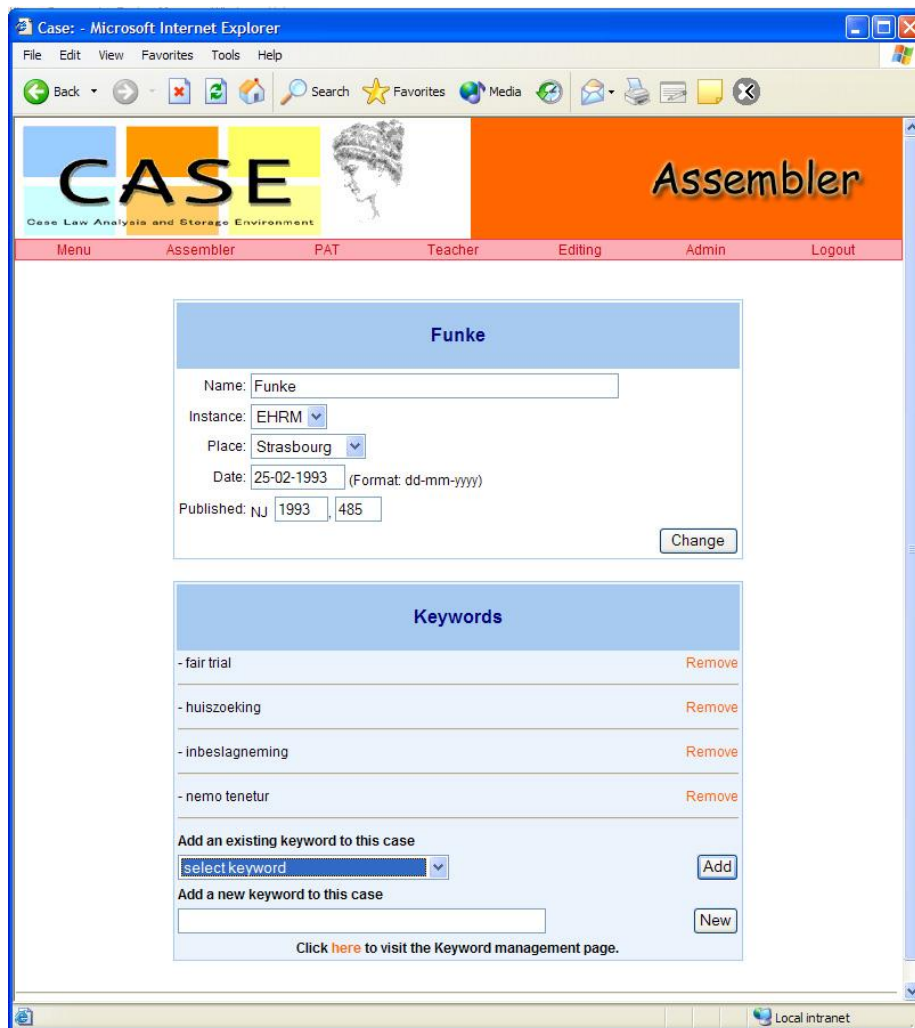


Figure 1: Metadata editor

The metadata editor interface is used to add or change metadata of a decision and, more importantly, to add new keywords, or remove existing ones. After completing this procedure, the decision can be searched for using the search interface.

The next step is the preparation of the decision for use. The prepare tool offers an interface that mimics the regular structuring and analysis interface: the editor needs to place pieces of text in the correct position within the solution framework (see figure 2).

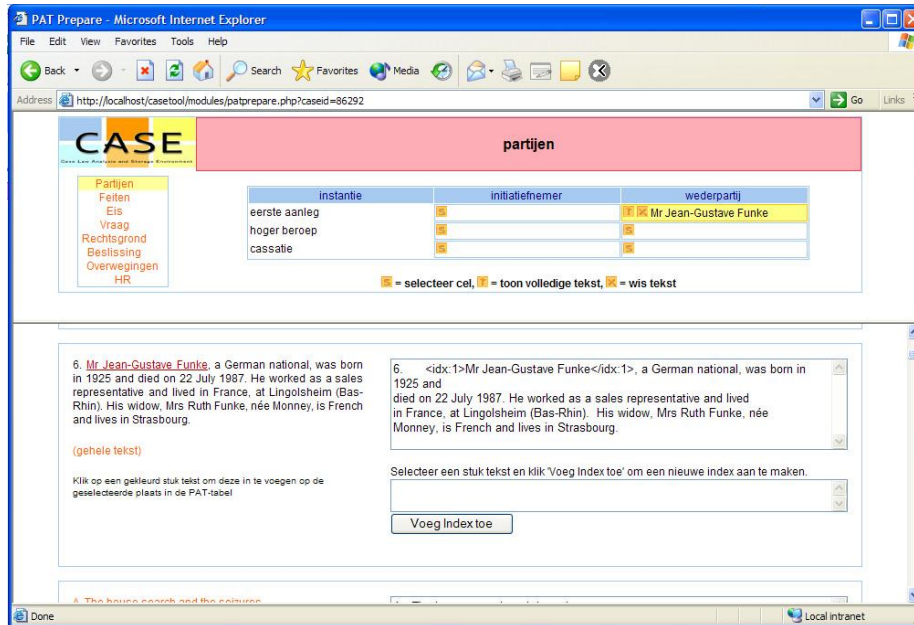


Figure 2: The prepare tool

Where the regular interface checks whether the correct text is in the correct position by consulting the database, the prepare tool writes the action of the editor to the database. The editor in a sense teaches the solution of the case at hand. Note that the editor does not have to add feedback to the database. Feedback is provided to the student in a case-independent way. When the teacher only wants part of the text fragment to be part of the solution, the editor can simply mark these smaller parts. This results in a text fragment with color coded sub fragments that can be placed in the solution table (e.g. Mr Jean-Gustave Funke in figure 3). After the editor has finished the above steps, the decision is ready for use by both teachers and students.

The teacher is not allowed to change the information or the solution framework of a decision.

However, he can add students to the CASE user database, and preview the correct answers (the prepared solution framework) for each decision. More importantly, the teacher has access to a student tracking facility to analyze student behavior.

This way the teacher can determine whether a student came to his or her end-result by simply trying every option, or by purposefully placing fragments in the solution framework.

Students can search the decision database using the search interface (see figure 3). This interface allows for metadata search – i.e. on publication date, publication place, court type, court location – but also supports Boolean keyword search and Boolean full text search. The student can also browse through all decisions in the database. The search result page offers support for associative search because key words and other attributes of the cases found are shown. The student can click on any of these to start a search on this attribute. Thus, for example, searching on all decisions with the same keyword of one of the decisions that were found by the original search is done by simply clicking on that keyword in the results page. From the same page, the student can print a decision or open it for structuring and analysis.

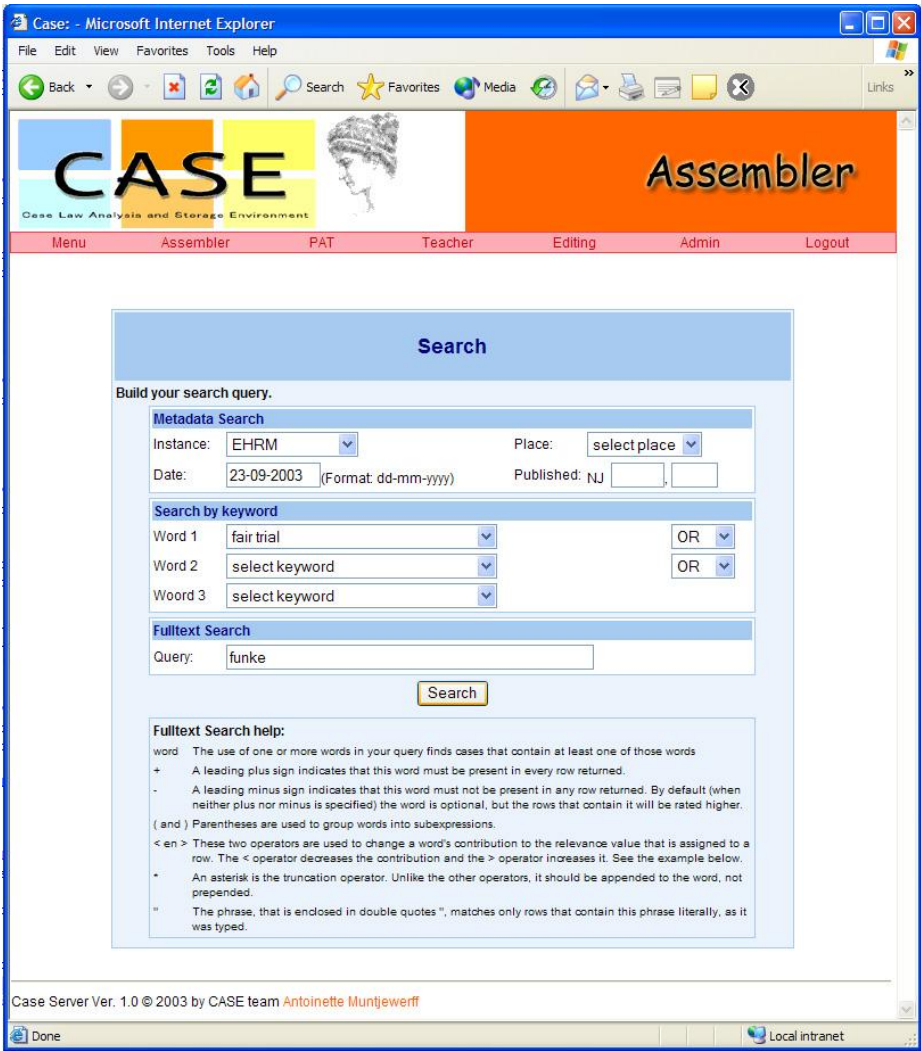


Figure 3: Search the database

The structuring and analysis interface, shown in figure 4, is divided into three frames. The left frame shows all text fragments of the decision at hand. The top right frame contains

the tables of the solution framework. The bottom right frame provides feedback to the student's actions. A text fragment is placed in a cell of the solution table by first selecting the cell, and then selecting the fragment to fill this cell. Once placed, the application will check the combination of cell and fragment and provide a feedback message from the database in the feedback frame. Text fragments can be removed from a cell by clicking the 'x'-button in the table. Once the student has placed all correct fragments in a specific table, she is notified of this through the feedback frame.

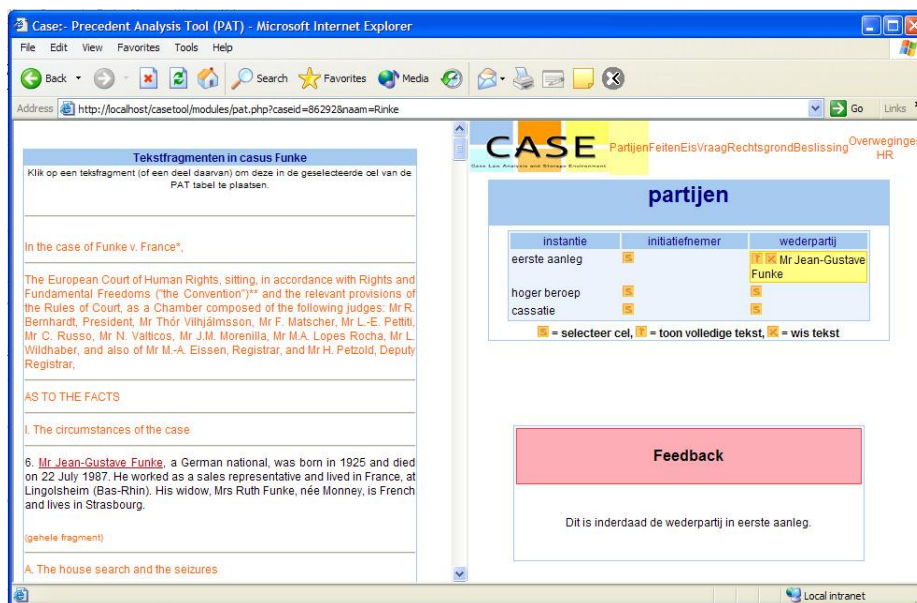


Figure 4: Structuring and analyzing a decision

VII. Summary and Future Work

Learning the law involves reading, structuring and analyzing decisions to be able to indicate the legal significance of the decision. Law students experience difficulties especially with determining what the decision adds to the body of applicable rules in the legal system. Within the current curriculum there is not enough time to read and analyze decisions in the presence of a teacher who may provide immediate feedback. Law students are also not presented with models that may guide them in the process of reading and analyzing decisions. In learning the law it is essential to know how to structure and analyze a decision.

CASE was designed to present the law student with an instructional environment in which she is able to analyze a decision in such a way that the structure is made explicit and the legal meaning can be extracted. CASE is implemented as a web-based server-side application model using open source software. CASE is easy to maintain and re-use and can

be made available in different languages. Future work involves testing the effectiveness of CASE. The claim that law students are supported by CASE in structuring and analyzing a decision in such a way that they are able to grasp the legal significance of the decision should be tested. The claim that it is easy to add a decision, to add key words and to prepare a decision for use should also be tested.

VIII. Bibliography

- Aleven, V. (1997) Teaching Case-Based Argumentation Through a Model and Examples, Pittsburgh: University of Pittsburgh.
- Ashley, K.D. and Brüninghaus, S. (2009) “*Automatically Classifying Case Texts and Predicting Outcomes*”, Artificial Intelligence and Law Vol. 17, No. 2, pp. 125-165.
- Breuker, J.A. and Van de Velde, W. (1994) CommonKADS library for expertise modeling. Reusable problem solving components. Amsterdam: IOS Press.
- Valente, A. (1995) Legal Knowledge Engineering. Amsterdam: IOS Press.
- Bos, A.M. (2003) “*Jurisprudentieleidraad*”, Encyclopedie der Rechtsgeleerdheid Vaardigheden. Amsterdam: Universiteit van Amsterdam.
- Eemeren van, F.H., Feteris, E.T., Grootendorst, R., Haaften van, T., Harder den, W, Kloosterhuis, H. and Plug, J. (1996) Argumenteren voor juristen. Oefenboek. Groningen: Wolters Noordhoff. (Argumentation for Legal practitioners)
- Ericsson, K.A., Charness, N., Feltovic, P.J. and Hoffman, R.R. (Eds.) (2006) The Cambridge Handbook of Expertise and Expert Performance. New York: Cambridge University Press.
- Franken, H. (1991) Practicum Methoden en Technieken II. Casusoplossen. Bundel ter dienste van het Practicum Methoden en Technieken van de faculteit der rechtsgeleerdheid van de Rijksuniversiteit te Leiden. Arnhem: Gouda Quint. (Solving Legal Cases)
- Haan den, N. and Sartor, G. (1999) “*Model-based Legal Knowledge Engineering*” Gaines, B.R. (Ed.). International Journal of Human-Computer Studies. Special Issue Model-based Legal Knowledge Engineering. Academic Press, pp. 1037 - 1042.
- Hage, J.C. (2009) “*Statische en dynamische rechtsfeiten.*“, Ars Aequi: juridisch studentenblad, 58, 91-98.
- Henket, M. and P. van den Hoven (1999) Juridische Vaardigheden in argumentatief verband. Groningen: Wolters Noordhoff. (Legal Skills)
- Jansen, L. (1999) Nieuwe introductie in het recht. Utrecht: LEMMA. (Introduction to Law)
- Keukens, W.T.M. and Nieuwenhuijzen van den M.C.A. (2008) Raad & Daad. Over de rechtsvormende taak van de Hoge Raad. Ars Aequi Libri, Nijmegen. (The Task of Judges)

Lubbers, A.O. (2007) *Belastingarresten lezen en analyseren*. SDU. (Reading and Analyzing Tax decisions)

Muntjewerff, A.J. and Groothuismink, J. (1999b) “*Supporting the Learning of Legal Case Solving using a Computer Program as the Instructional Environment*”, Proceedings CATE’99 Computers and Advanced Technology in Education. Philadelphia, USA: IASTED.

Muntjewerff, A.J. (2000) *An Instructional Environment for Learning to Solve Legal Cases*. PROSA. Amsterdam: University of Amsterdam.

Muntjewerff, A.J. (2002a) *HYPATIA research program. Models of Legal Knowledge and Legal Reasoning in Electronic Materials for Learning the Law*. Amsterdam: University of Amsterdam, ISBN 90-806969-1-9.

Muntjewerff, A.J. (2002b) “*Principled and Structured Design of Electronic Materials for Learning the Law*”, Bench-Capon, T.J.M., Daskalopulu, A. and Winkels, R.G.F. (eds.), *Legal Knowledge and Information Systems. Jurix 2002: The Fifteenth Annual Conference*. Amsterdam: IOS Press, 2002, pp. 133-142.

Muntjewerff, A.J. (2002c), “*Evaluating the Instructional Environment for Learning to Solve Legal Cases PROSA*.”, Torrellas, Gustavo A. and Uskov, Vladimir (eds.), *Proceedings of the IASTED International Conference Computers and Advanced Technology in Education*. Anaheim Calgary Zurich: ACTA Press, pp. 374 - 379.

Scholten, P. (1974) *Algemeen Deel*. Zwolle: W.E.J. Tjeenk Willink. (General Part. On Legal Decisionmaking).

Soeteman, A. and Wolthuis, A.J. (2003) *Syllabus Juridische argumentatie 2003*. Amsterdam: VU. (Legal Argumentation)

Stolwijk, S.A.M. and Bosch, M. (2002) “*Het bestuderen van arresten in strafzaken*”, *Arresten strafrecht/strafprocesrecht met annotaties ten behoeve van het onderwijs verzameld*. Deventer: Kluwer. (Studying Legal Decisions in Criminal Cases)

Address: Dr. Antoinette J. Muntjewerff University of Amsterdam, Faculty of Law, Amsterdam / The Netherlands. E-mail: muntjewerff@uva.nl