Legal Knowledge Based Systems

JURIX'96

Foundations of legal knowledge systems

The Foundation for Legal Knowledge Systems

Editors:
R.W. van Kralingen
H.J. van den Herik
J.E.J. Prins
M. Sergot
J. Zeleznikow

Paper version: Tilburg University Press, ISBN 90-361-9657-4, NUGI 699

CONTENTS

A TASK-BASED HYPERINDEX FOR LEGAL DATABASES

Luuk Matthijssen
Center for Law, Administration and Informatization[#]
Faculty of Law, Tilburg University
P.O. Box 90153, 5000 LE Tilburg, the Netherlands
email: L.J.Matthijssen@kub.nl

Abstract

This paper addresses the problems that lawyers experience retrieving information from legal-text databases. Traditional access mechanisms of text databases require users to know how information is stored. I will propose a method for index organisation which shields lawyers from the internal storage structures and which allows them to address the legal databases in their own legal terms. The proposed index is based on a model of legal tasks as opposed to traditional database indexes which represent the contents of the database. I will lay out the architecture of an information system in which this taskmodel is used to determine the information need, to retrieve relevant documents and to give methodical guidance for the legal task itself. To account for the design of a task-based legal information retrieval system, a substantial part of this paper is devoted to analysis and representation of legal tasks.

1 Introduction

Lawyers in their work have to consider ever growing amounts of information. Despite efforts to deregulate, the volume of statutes and treaties is still growing. Also the body of recent case law is expanding because society asks for legal decisions more often. Information retrieval (IR) technology can help store and manage this textual information in legal databases. Text databases can traditionally be accessed through an index based on the contents of the database, much like the indexes of books. Indexes can be browsed (as in books) or queried (a specific database functionality). In this paper, it is argued that indexes based on the contents of the database fall short for large text databases. As the information represented in the index is based on the contents of the database, the user has to be familiar with the content and the structure of the database to find his way in the index. In contrast, an index representing the user's task domain can make it easier for the user to choose browsing steps and to formulate queries for they can be put in his own terms (*i.e.*, the terms of the task or the problem at hand). Besides providing access to the database this taskmodel can also give methodical guidance for the task itself. It can operate as an intelligent intermediary between the lawyer and the database.

In section 2 of this paper the problem of legal text retrieval is explored and a design for a task-based hyperindex system is presented. In section 3, argumentation theory is used to analyse legal tasks both in general and specifically for one legal task from the domain of the Dutch law of administrative procedure. This example domain is used in section 4 to develop a task-based hyperindex for. Also in this section, it is shown how this index is used for methodical task support and for the retrieval of legal documents. Section 5 pays some more detailed attention to the information retrieval mechanism. The conclusions are presented in section 6 as well as the projected follow-up of this research project.

[#] This research is carried out in cooperation with the Department of Computer Science of Eindhoven University and is funded by the SOBU foundation.

2 The problem of legal text retrieval

2.1 Legal text retrieval

The texts of Dutch statute law and selected case law are stored in legal databases which are published on-line and on CD-ROM.[1] A lawyer experiencing an *information need* while working on a specific legal problem can consult documents in the database via an index. The *index* of a database consists of a list of *terms* (words and word combinations) that characterise the contents of the database and refer to the stored documents much like a book's table of contents and subject index.

There are generally two ways to access the database through the index: *browsing* and *querying*. You can browse through the list of available index terms to find one that expresses your information need and request the corresponding documents. Another way to retrieve documents is by formulating a query q to express the information need N (see Figure 1). The query language L generally allows search terms and boolean operators. The query terms are matched with the index terms and the corresponding documents are retrieved. Some query languages allow natural-language and graphical queries.

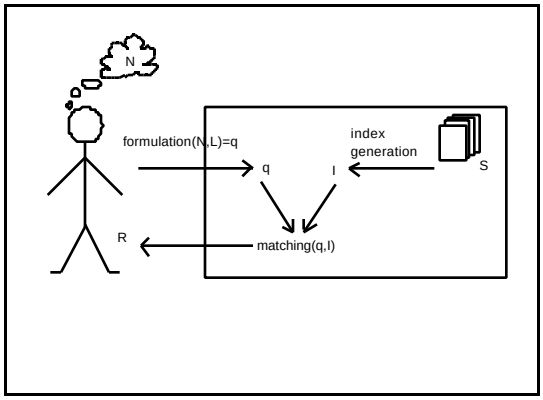


Figure 1: Retrieval of documents from text databases.

Just as there are various query languages, there are several indexing methods and systems to relate a query to an index (*e.g.*, boolean, vector-space, probabilistic). I will not discuss the details of all these systems. For the purpose of this paper I will focus on one aspect of legal information retrieval which these systems have in common.

> *A lawyer searching for information on some specific legal problem has to specify terms that distinguish the documents that are relevant to his information need from the other documents in the database.*

For lawyers this task is all but straightforward and it poses problems which stand in the way of finding the relevant documents. The problems that users face while addressing text databases are generally recognised and well described in literature (Salton, 1989). At the root of these problems there are three key conditions of the basic text retrieval model which most often are not met. For the optimal use of text retrieval systems the users must:

- know and be able to articulate their information need; and
- know the content and the storage structure of the documents in the database; and
- $ be aware of the operation of the retrieval mechanism.

[1] An important development in this respect is the recent introduction of a CD-ROM database containing the full text of Dutch statute law, the *Algemene Databank Wet- en Regelgeving (ADW)* by Kluwer Datalex.

In other words: lawyers are expected to translate the information need they have in mind in the form of legal concepts into a query which should be put in technical database concepts. I have referred to this obstacle in an earlier publication as the *conceptual gap* (Matthijssen, 1995).

For lawyers it is difficult to translate an information need occurring in the context of a legal problem into a database query. It is hard to tell beforehand what information is required to solve a legal problem; only lawyers themselves can make this assessment once they have found the documents. Also, lawyers lack an inside view into the storage structures of the database and the operation of the retrieval system. As a result most of the context information that determines the information need is lost in the query formulation process and the resulting query is too unspecific.

The browsing method of addressing the database helps lawyers explore the available subjects and explore their information need. However, as the index reflects the contents of the database they are still required to be familiar with the organisation of the documents in the database. I will propose a different method for index organisation to shield lawyers from the internal organisation of the database and the operation of the retrieval mechanism and to help them provide the context of the problem which is necessary to specify the information need. In the following I will first discuss some related methods for index organisation.

2.2 Index organisation

Enhancing the browsing facilities is one way to relieve the user of the difficult query-formulation task. Browsing facilities of an unordered index however are limited. Because of the size of the databases and the variety of subjects covered by the documents, the index lists can be very long and the terms required to describe an information need often occur dispersed across the index.

One way which is used to facilitate browsing is *thematical index organisation*. In a thematically organised index related terms are grouped together. Although the grouping of terms is arbitrary, this can reduce the effort of finding that particular term which characterises the information need best. An extension of the thematical index organisation is hierarchical index organisation.[2]

In a *thesaurus*, index terms are linked to synonyms and related terms. The terms of a query are automatically extended with linked terms to retrieve a larger set of associated documents. Thesaurus search is often described as a form of dynamic query extension. It can also be considered as an extension of the index. If the thesaurus is used as a query-extension method the query-formulation problem remains unsolved. If the thesaurus is used to facilitate browsing it also provides poor aids for finding the relevant documents. Organisational links are still based on the contents of the documents in the database. As legal databases contain documents on a wide variety of legal issues this provides a very non-specific context. Moreover, you have to be familiar with these contents to be able to find your way.

An index organisation method of growing importance is the *hyper-index*, in which the index is organised as a hypertext. Hypertext is based on the idea that, since the human mind works associatively, it is best to represent information in an associative manner.[3] In a hyper-index, related terms are connected through links which can be followed automatically. In my view, hypertext navigation is an unsuited mechanism for information retrieval. It is true that the human way of searching and exploring an information need is associative but these associations are not trivial; they have intended semantics. The purely generic associative links of a hyper-index lack these semantics and that is why following these links most often does not correspond to users' intentions, resulting in

[2] For examples of hierarchical or layered index organisation see (Bruza, 1993) and (Agosti et al., 1991).
[3] For an overview of the principles and possibilities of hypertext see (Conklin, 1987).

disorientation. Although, I do recognise the importance of hypertext as an interface technique especially in comparison with query interfaces, I would like to emphasize the need for typed links and careful design of the index, based on thorough task analysis.

2.3 Task-based index organisation

The solution to bridge the conceptual gap proposed in this paper, considers the organisation of the index. If the index terms and the organisational links connecting them are made to correspond with activities and methodical steps of legal tasks the lawyer can address the database in his own terms. Notice that the representational focus of the index is reversed: instead of modelling the contents of the database the index represents the task domain of the lawyer. This means that the retrieval system is dedicated to one specific (type of) legal task and for each different type of task we will have to build another index. The database however remains generic and can be used without alterations for any type of legal task.

By supplying the index with knowledge of the lawyer's task domain we can make it operate as an *intelligent intermediary* between the lawyer and the database, delivering relevant documents at the right moment and giving guidance and advice for the legal task. While performing a legal task the lawyer follows a path through the network guided by the framework of the task-based index. This path through the network can thus be seen as a representation of a specific legal problem from which the co-occurring information need can be derived. The domain model provides the context information necessary to make a specific expression of the information need. With this extended expression of the information need the relevant documents can be retrieved from the database at the moment they are required. Apart from the problem of expressing an information need, also the navigation problem is resolved. The task-based index has well-defined term and link semantics which comply with the methodical steps of legal tasks as lawyers perceive them. A location in the index can always be related to the problem or the task at hand preventing the disorientation that can occur while browsing in generic hyper-indices.

The conceptual gap between the lawyer and the index can thus be bridged. However, at the same time another gap, that between the index and the documents in the database, is widened. Index terms are no longer statically linked to documents. This means that we have to find another way to link the expression of the information need to relevant documents in the database. For this I will propose a retrieval strategy using dynamic linking.

2.4 Task-based hyperindex system

In the research project this paper reports on, a prototype of an information system using a task-based hyperindex is developed for a specific legal task.[4] Before describing this system in more detail I will outline the systems architecture. See also Figure 2.

[4] This prototype is called ARMOR which is an abbreviation for ARgument MOdel based Retrieval system. The interface part of the system is made in Asymetrix Toolbook and the database part is made in Folio Views.

The systems domain knowledge is represented in the *Task model* which has the form of a hyperindex. As mentioned before the case representation has the form of a path through the hyperindex network. This sequence of textual expressions is stored in the *Project database*. There are several *Legal databases* containing the full text of statutes and case law. A *Retrieval manager* provides for the communication with these databases through the execution of internally generated queries and the propulsion of retrieved documents.

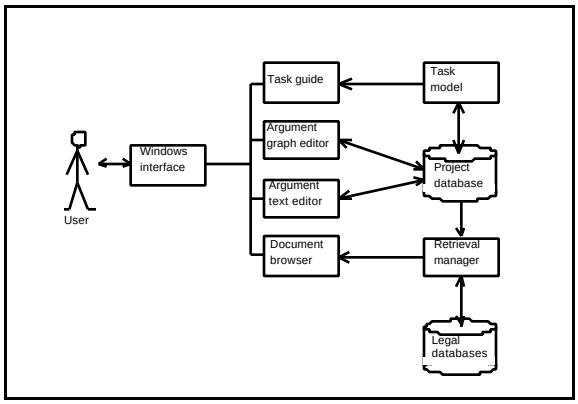


Figure 2: Component modules of the task-based retrieval system.

The information support functions described in the previous section are performed by four different system modules. Although these are distincly different functions they are presented in a single Windows interface. Users are presented with two views on on the Project database through which the project database contents can be manipulated. The *Task graph editor* presents a graphical display of the case representation and allows direct manipulation of the structure of the task elements and their interrelations. The *Task text editor* is used to edit the textual expressions making up the contents of the case representation. These activities are supported by methodical guidance through the *Task guide* and information support from the *Document browser* which presents documents retrieved from the legal databases.

3 Analysis of legal tasks

3.1 Argumentation

The focus in this research is on the retrieval of information from legal databases. I have proposed a taskbased hyperindex as an intelligent intermediary between these databases and the lawyers using them so they can address the databases in their own terms. To this end the hyperindex must be made to represent the lawyers taskdomain. In the following I will develop a network model which can operate as a hyperindex for a specific legal task. To build such a network model for a specific legal task we have to analyse legal tasks as to determine the buildingblocks of the network model and to establish the legal semantics of the network elements. In the next sections we will discuss the IR function of the network model as an index. In this section we focus on the task support function of the network model and on the way it represents a legal taskdomain.

The field of science which investigates the mechanisms of legal decision making as a process and which tries to establish principles for valid or convincing decision making procedures is the theory of argumentation. As far as the structural analysis of legal reasoning is concerned, the theory of argumentation is akin to logic theories.[5] There are however distinct differences. Logic solely focuses on the structure of reasoning. If the premises are valid and the reasoning pattern is syntactically sound then the conclusion

[5] Toulmin in his initial publications and Perelman have seriously questioned the use of logic for the analysis of legal reasoning. In his recent publications Toulmin puts more emphasis on the close relation between argumentation and logic. See (Toulmin, 1992).

is valid. These reasoning patterns however do not comply with the reality of human reasoning especially in moral issues. They can at best give an artificial and idealised model of reasoning. According to MacCormick (1992) logic only approaches reality for reasoning in clear and simple cases. Argumentation theory on the other hand addresses the reality of human problem solving in moral issues. Decisions are not qualified as true or valid but as convincing or relatively appropriate (Voermans, 1995).

In argumentation a distinction is made between the *process* of legal decision making and the *result* of this process, a motivated legal decision. Because argumentation theory considers the decision making process it is better suited for the modelling of legal tasks than logic.[6]

Two influential contributors to the theory of argumentation are Perelman (1969) and Toulmin (1958). Toulmin describes the justification of a legal decision as a procedure consisting of a fixed order of steps. These reasoning steps also constitute the elements that are put forward in the motivation of the decision. Argumentation in this model is acceptable if it is constructed according to the prescribed procedure and made up of the corresponding necessary elements. These formal requirements are domain-independent and therefore applicable to legal reasoning as well as to, for example, scientific, managerial and ethical reasoning. Toulmin admits that ultimately formal criteria alone are not enough to make argumentations acceptable. That is why he also distinguishes criteria regarding the contents of arguments. Criteria for the contents of arguments by their very nature are domain-specific. Notably in the legal domain there are specific customs and more or less accepted norms regarding the contents of arguments.

I will use Toulmin's argumentation theory as a starting point to build a legal taskmodel as it offers handles to describe both the activities and the result of legal reasoning and it differentiates between general and domain-specific requirements.

The steps of the Toulmin model correspond with the different kinds of critical questions that can be asked for any motivated position statement. The six critical questions are:

- What exactly is the main *claim* in this position?
- What *grounds* are put forward to defend this claim?
- How can the step from these grounds to the claim be *warranted*?
- Is there sufficient *backing* for this step?
- How strong is the claim? (characterise the step with a *modal qualifier*)
- What possible *rebuttals* apply to this argument?

Most often the analysis of a position statement requires more than the linear, one-time application of these steps. In legal practice there often is no agreement as to the legal qualification of grounds and to the applicability of rules as warrants. These steps can be analysed as claims in their own right requiring a separate motivation. The critical analysis thus results in a layered argumentation network of which the structure can be analysed to assess the argumentation. Figure 3 shows an example of such an argumentation network. For simplicity the optional qualifiers and rebuttals are left out.

[6] This does not mean that I disqualify logic for applications in the legal domain altogether. There are certain principles regarding the structure of legal decisions which can well be described by logic, especially with extended logic representations such as proposed by Prakken (1993), Hage, Leenes and Lodder (1994) and Royakkers (1996).

The idea to use Toulmin's argumentation theory to formalise legal reasoning for the application in legal knowledge-based systems is not new. Several other authors have discovered it's powerfull analysis framework.[7] They have proved Toulmin's theory gives good handles for the formalisation of legal reasoning in discretionary domains.
In this research Toulmin argument structures are used as buildingblocks for the modelling of legal tasks in a hyperindex. In the following sections a hyperindex is presented for a specific legal task: the objection procedure in administrative law. I will show how the argumentation model applies in this domain and how it can be used to relate the statutes and case law documents that are stored in legal databases, to this specific legal task.

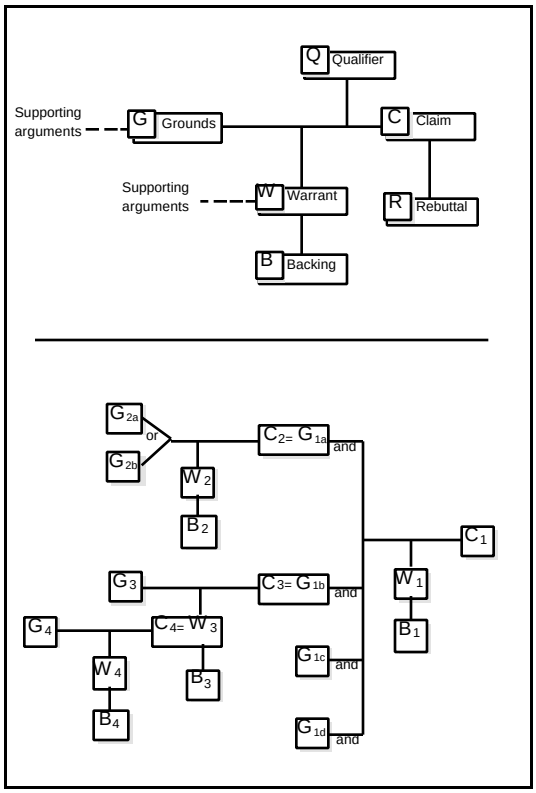


Figure 3: Simple Toulmin argument structure and example of a composite argumentation structure.

3.2 Objection procedures in administrative law

To test the proposed task-based index for legal IR, in this research project a prototype is being developed for a specific legal domain. The domain that was chosen for this purpose is the Dutch law of administrative procedure. Administrative law concerns the relation between the government and its citizens. In the Netherlands this field of law is governed by the 'Algemene wet bestuursrecht (Awb)' (General Administrative Law Act). The Awb is a general law specifying the procedural framework and constraints for specific administrative laws. It comprehends procedures for decision making by administrative agencies and procedures for citizens to file objections if they do not agree with an administrative decision.

The Awb demands from administrative agencies that their decisions are prepared with proper care, that they are well-motivated and that both the decision and the motivation are put in writing so the citizens whom it concerns can assess the decision making process and the grounds for the decision. This provides them with the first information they need to object to the decision if they do not agree with it.

The Awb dictates an administrative pre-procedure which must be followed before the case can be taken to court. Interested parties who do not agree with the decision first have to try and resolve the conflict with the administrative agency which has made the decision. To this end they can file a letter of objection and ask the administration to

[7] For applications of Toulmin's argumentation theory see Dick (1991), Bench-Capon (1995), Voermans (1995) and Zeleznikow (1995).

reconsider the decision. Objection procedures too are constrained by the motivational and procedural requirements of the Awb.

The Awb is only a part of the normative framework for governmental action. Administrative agencies must also observe the general principles of proper administration which have been developed in the practice of administration and administrative law. These principles have been well studied and documented in legal literature and they are often applied and referred to in case law. The general principles of proper administration have also played an important role in the design of the Awb to the extent that some of the principles are incorporated in the rules of the Awb. Apart from these general rules there are many specific administrative regulations for the different fields of government policy. These regulations specify the general rules of the Awb as to address specific administrative agencies and to provide grounds for protecting specific interests.

The legal task that we chose for to design a prototype of a task-based IR system, is the objection procedure of the Awb. It is in many respects typical of a legal task for which information from legal databases is required and which can be characterised as a complex argumentation process. A peculiarity of this task is that often objections are filed by legal laymen while writing a good letter of objection requires legal knowledge. This is an interesting side issue for it gives good purpose to the methodical guidance and the information support the taskmodel can offer.

If we analyse the objection procedure with respect to the parties involved it helps to consider Perelman's view on argumentation. To choose and design an argumentation it is important to consider the audience that is addressed. In objection procedures there are two parties directly involved. The party filing the objection must not only lay out arguments to convince the administrative agency, it must also counter the arguments of the administration's decision. Furthermore, the two parties in their dialogue must at all times consider the silent presence of a much wider audience. They must observe the rules of the Awb and provide generally convincing arguments because there is always the possibility that the case will eventually be presented to court.

4 Hyperindex and taskmodel

Our focus domain is the task of objecting to an administrative decision. In this domain the Toulmin argument scheme applies in two ways. First to analyse the administrative decision and second to construct a letter of objection countering this decision. These two main tasks, for the purpose of specifying the taskmodel, have been broken down further into their constituent activities with input/output and precedence relations.[8] The activities in the resulting detailed procedural taskmodel are related to relevant documents by means of argumentation relations. An outline of the activities in the procedural taskmodel is displayed in the project management screen of Figure 4. In the following, the procedural model and the argumentation relations which link to relevant documents are discussed for the two main tasks.

[8] For a description of this model refer to the activity analysis presented in another report on this research. (Matthijssen, 1996)

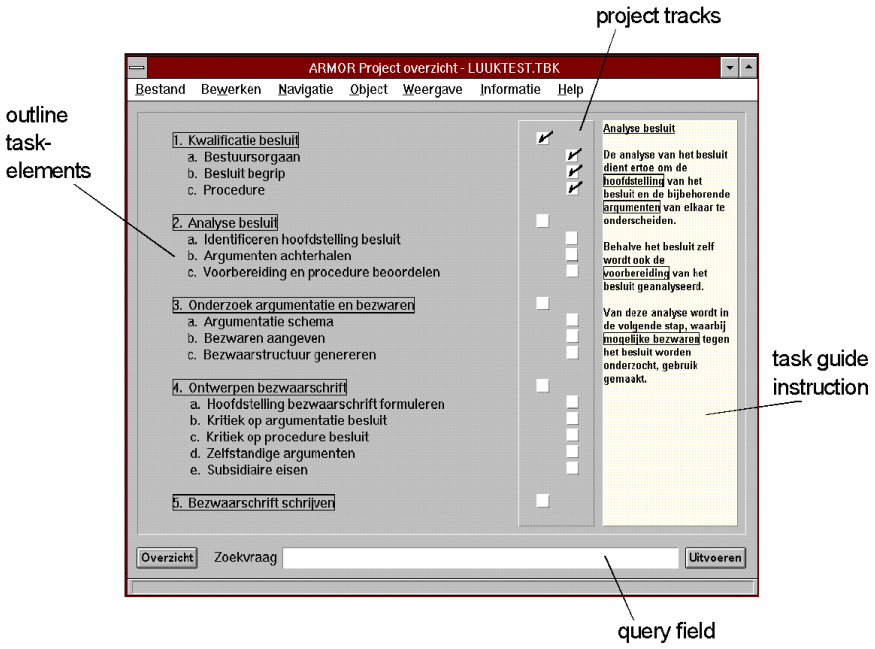


Figure 4: ARMOR prototype: project management screen.

For this discussion an example of a specific objection procedure is used which is presented in the following and all the subsequent text boxes.

> Someone wants to file an objection to a decision of the Burgomaster and Aldermen of his city revoking his building permit. The decision of the Burgomaster and Aldermen is supported with the argument that the building activities which have already started are violating the conditions of the building plan for which the permit was issued.

4.1 Analysing the administrative decision

Citizens who are confronted with an administrative decision, according to the Awb are notified by means of a written document. This document is the first and best lead to find relevant information and to formulate objections. In successive steps the user can analyse the administrative decision. Purpose of this exercise generally is to prepare for possible objections. The claims and arguments (and possibly other elements of the Toulmin model) of the decision are identified and the statutory context and the objection procedure to be followed are determined. Administrative decisions must always contain (a reference to) the specific law which grants the administrative agency the authority for the decision. As these specific laws generally also state how this authority is to be used we can use this data to determine the statutory context of the decision. Analysing the administrative decision is to a large extent a classification task. The case model for this task is built up in a sequence of classification steps which are presented on the screen of Figure 5.

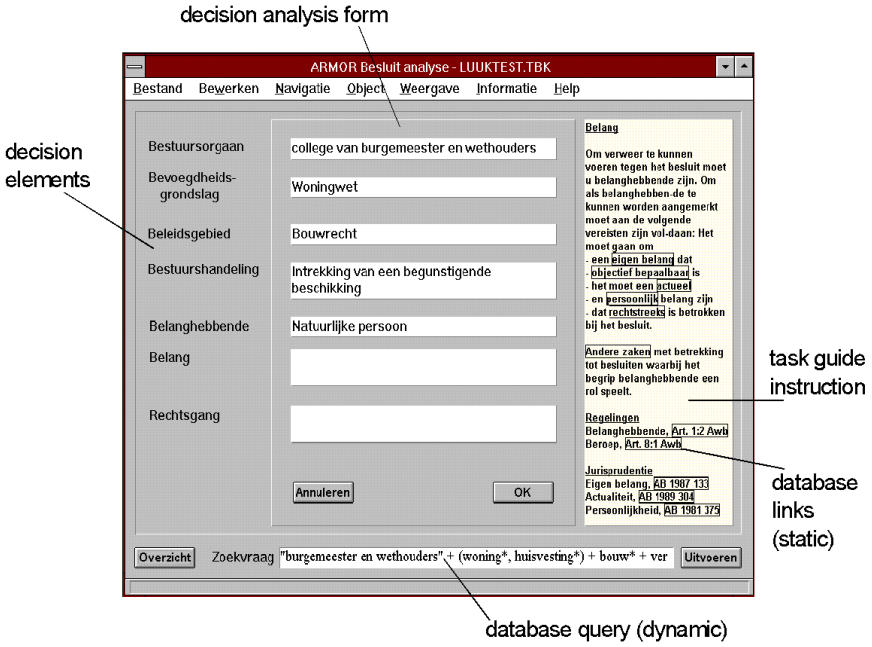


Figure 5: ARMOR prototype: decision analysis screen.

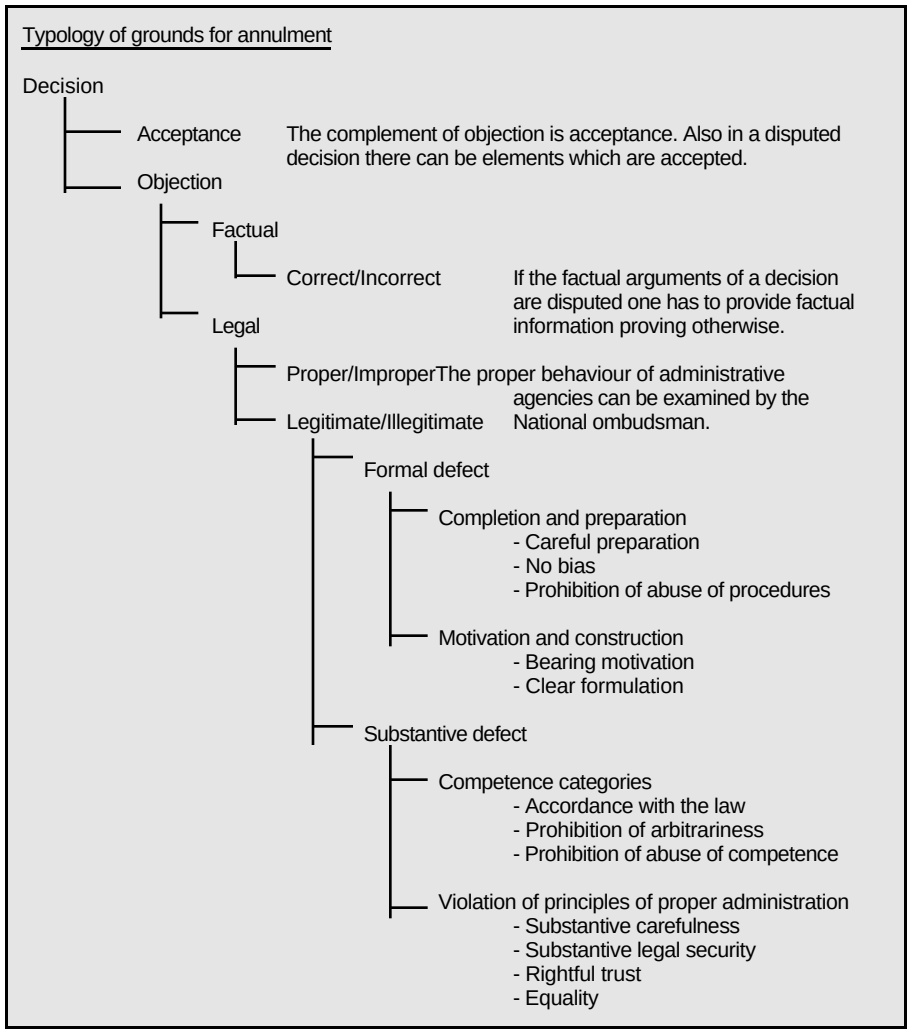
This screen also shows the instructions of the task guide for the active decision element which consist of explanations on the possible categories and links to relevant Awb provisions. Once (a part of) the case model has been built, it can be used to retrieve relevant case law and specific administrative laws from the database. The query field shows the database query which is generated automatically from the case model. In section 5.2 of this paper this dynamic linking mechanism is described.

The administrative agency that revoked the building permit is the Court of Burgomaster and Aldermen which is classified as *an agency of a legal person under public law*. This classification is deciding for the applicability of Awb norms. The type of decision is *the reversal of a favourable decree*. The interest of the person losing his permit is a direct interest. The field of policy is *Building and housing*. The main regulations granting the Burgomaster and Aldermen authorities in this field of policy are: the Housing Act, the Town and Country Planning Act and the Building Materials Soil Protection Decree. Article 59 of the Housing Act is also mentioned in the administrative decision as the ground for the decision making authority. From this complex of facts the task guide derives that an appeal can be made. Before the case can be presented to court first an objection must be filed with the agency that has made the decision.

4.2 Objecting to an administrative decision

Once the administrative decision is analysed and classified and the right procedure is determined we arrive at the second task: objecting to the decision. The objection consists of a claim which counters the claim of the administrative decision and a set of supporting arguments. As the purpose of the objection is to counter the administrative decision a stronger position is built if also the administrations arguments can be countered. For the legal type of arguments (as opposed to the factual type) the most authoritative grounds are statutes and case law which will need to be retrieved from the

database. The construction of arguments is supported using the systems taskmodel which comprises a typology of possible grounds for annulment. Grounds for annulment of administrative decisions are the logical complements of the general principles of proper administration.



The terms in this typology (of which the meaning may be a little obscured by their translation) all represent categories of considerations to which administrative decisions must comply and to which they can be tested. They are also possible grounds for objections to these decisions. Together with the procedural taskmodel the typology specifies at what moment in the process these considerations are made. The categories also differentiate as to the type of arguments which are required to be retrieved from the legal database.

The building permit is revoked on the grounds that the building activities do not comply with the conditions of the building permit. In this case there is disagreement as to the facts of the case. To substantiate this argument, factual grounds must be presented. The task guide advises to further investigate if the administrative agency was sufficiently efficient in gathering information as to live up to the principle of careful preparation. If not, a general principle of proper administration has been violated. The grounds for this argument are provided by the decision itself, Awb regulations and case law.
If a decision has substantive defects such as violation of the law, it violates some specific administrative law (often rules of competence). The competence for making a certain decision fails, is insufficient, or is falsely applied. The grounds for this type of argument is therefore to be found in the competence provisions of the specific administrative law.

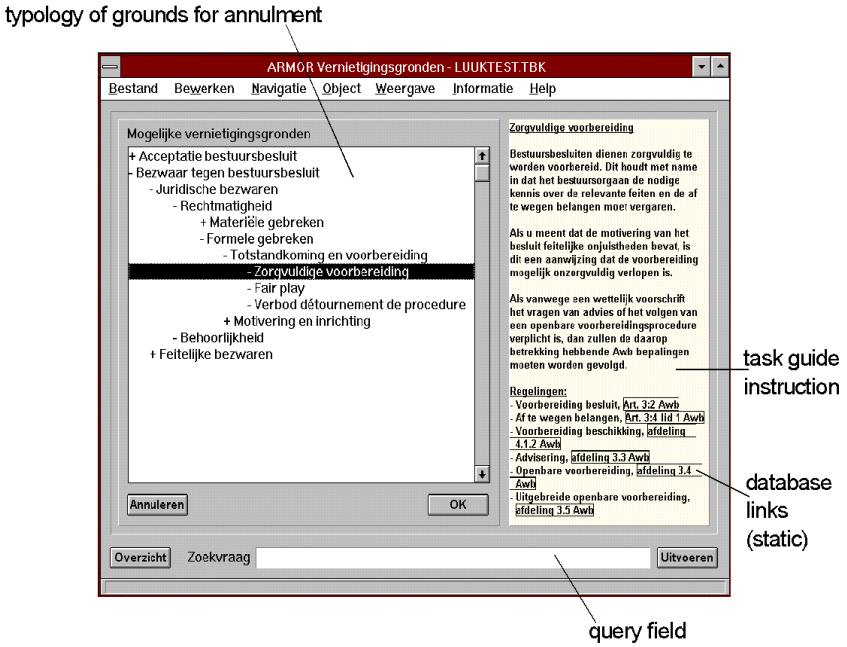


Figure 6: ARMOR prototype: Argument selection screen.

In the prototype the typology of grounds for annulment of administrative decisions is used to identify the contents of objections to be able to link the arguments in the case model to the documents in the database supporting them. As the screen shot of Figure 6 shows the taskguide presents advise on how to use and support specific types of arguments. For each type of argument it provides links to relevant Awb provisions. Case law and specific administrative regulations are retrieved by means of the dynamic linking mechanism described in section 5.2. Once the arguments have been selected the case model is forwarded as a document template to a word processing application for further editing and completion of the letter of objection.

5 Information Retrieval strategy

The starting point for information retrieval is the user's information need. Indications for the information need can be divided into structural and content indications. Structural indications of the information need consist of the methodical steps taken in the argumentation framework of the task model. The content indications are represented by the typology and the text of the arguments. As far as the argument structures correspond to the task model a set of *static links* to relevant documents is presented by the task guide. Users can also extend these predefined structures and add their own content to the selected argument types. If the contents of these extended arguments is used as an indication of the information need, links cannot be predefined and a *dynamic linking* mechanism is required. This dynamic linking mechanism uses search terms that are assigned in the task model to specific argument types and other contents indicators such as the words used in the text of the arguments. The case model thus maps into a combination of search terms (a database query) which links to documents in the database.

5.1 Static linking strategy

In order to build an argumentation for both the analysis of the administrative decision and the objection to this decision, lawyers can use predefined argument types. Part of the information need occurring while filling in the text of these arguments, can be determined beforehand. To the extent to which the context is known the case model can provide links to relevant documents in the database. These links are static because they are appointed at modelling time. Used in this way the taskmodel has a similar function as the hyper-indices discussed in section 2.2.

> In the analysis of the administrative decision the task guide presents links to motivational requirements in articles 4:16 to 4:18 Awb. These are sources of possible grounds and backings for counter arguments. For the assessment of the decision procedure a link is supplied to article 4:14 which prescribes the notification demands and the terms (in the sense of time constraints) for administrative decisions. To help check the contents of arguments in administrative decisions there are links to articles 3:2 to 3:4 Awb for they demand that decisions are prepared with "proper care" and careful weighing of interests. These links are accompanied by links to case law that helps explain and interpret the notion of "proper care". For writing the main claim of a letter of objection the task guide provides a link to article 6:5 Awb which specifies some general requirements for the contents and the components of the letter.

5.2 Dynamic linking strategy

The procedural taskmodel can only indicate the need for general information concerning motivational and procedural rules and guidelines. To be able to provide also specific and substantive information on the subject of case we must consider the contents of the arguments. While following a path through the tasknetwork the user makes selections and writes the texts of the arguments which constitute the letter of objection. These texts are scanned for words and word combinations which indicate objections from our typology. If such objections are recognised together with the context of the activity node in which they occur they give a good lead on the relevance of documents in the database.

The following table shows a path fragment of the example that was introduced before.[9] It considers the phase in which the administrative decision is analysed. The rows in the table from top to bottom represent a sequence of activity nodes making up the case representation. The text constructed in the activity nodes is scanned for words and word combinations which indicate a the subject of the text. To this end, for each activity node we use a context specific domain vocabulary. The keywords in these vocabularies because of their specific context and their specific meaning are good indicators of the subject of the text and also of the co-occurring information need. The keyword column shows the recognised keywords from the text, the subject column displays the subject names and the documents column displays the terms describing the co-occurring information need. These retrieval terms are combined into database queries on the basis of the types of activity nodes and the links between them. (The NEARX operator specifies the maximum number of positions (X) between two terms in the document text.)

Activity sequence	Text keywords	Subject	Documents
Legal domain	Building and housing	town development, building permit, development plan	"General Administrative Law Act", "Town and Country Planning Act"
Adm. agency	Burgomaster and Aldermen	Burgomaster and Aldermen, competence	"Burgomaster and Aldermen", can, competen*
Administrative act	Revoke building permit	reversal of beneficial decrees, building permit	permit NEAR2 build*, permit NEAR6 (revok* OR anull* OR cancel*)
Main position	building permit ... revoked	...	...

To retrieve the relevant legislation the path representation is translated into a database query. Possible queries from the example in pseudo-SQL are:

```
SELECT "Article" FROM
"Town and Country Planning Act"
WHERE
TEXT = "Burgomaster and Aldermen" NEAR6 (can OR competen*)

SELECT "Article" FROM
"General Administrative Law Act" OR "Town and Country Planning Act"
WHERE
TEXT = (permit NEAR2 build*) OR (permit NEAR6 (revok* OR anull* OR cancel*))
```

Another path fragment with objection arguments is shown in the next table. This particular line of argumentation focuses on the costs already made and the confidence that was inspired when the building permit was first granted. The argument numbering conventions, also explained in the table, are adopted from van Eemeren and Grootendorst (1992).

Activity sequence	Text keywords	Subject	Documents

[9] Because the retrieval system uses the Dutch language this retrieval example in English is constructed by hand.

Claim 2	Building permit should be sustained	object to reversal of beneficial decree	"Case Law Administrative Courts", "beneficial decree" NEAR5 (revers* OR revok*)
Argument 2.1 (first argument supporting claim 2)	confidence in validity of building permit	rightful trust	"Case Law Administrative Courts", "rightful trust", "principle of trust"
Argument 2.1.1 (first argument supporting argument 2.1)	costs already made	damages upon trust	"Case Law Administrative Courts"
Grounds 2.1.1 (grounds for argument 2.1.1)	progress building activities	factual information	"General Administrative Law Act"

Some possible queries from this second path fragment are:

SELECT "Article" FROM
"Case Law Administrative Courts"
WHERE
TEXT = "beneficial decree" NEAR5 (revers* OR revok*)

SELECT "Article" FROM
"Case Law Administrative Courts"
WHERE
TEXT = ("rightful trust" OR "principle of trust") AND ("beneficial decree" OR permit)

One path translates into several queries all of which deliver different document sets. This approach is easier than devising one very complex query which has to deliver all the relevant documents. It also makes way for relevance ranking of the retrieved documents. Documents occurring in more than one document set are ranked higher than documents occurring only once.

The algorithm currently used to translate path representations into database queries uses standard phrases specified in the taskmodel, for clusters of steps such as a claim and it's supporting arguments. We are also looking into different algorithms for path interpretations to see which ones work best. A path can alternatively be interpreted as a cooling down trail (the last node is the most important one), a chain of assertions, or a chain of assertions and negations (unchosen option are interpreted as negations). See the work of Berger (1995) for possible ways in which paths through a hyperindex can be interpreted.

6. Conclusions and further research

The status of the research project which this paper reports on is the on going development of a prototype. In the current prototype of the system the complete architecture and the taskmodel are implemented. We are now completing the retrieval mechanism so we can start testing the retrieval performance. Tests are scheduled with domain experts to validate the knowledge models. After that we will be testing the system with laymen in the field of administrative law to verify the methodical guidance of the task framework. As the systems development method of prototyping is adopted, we intend to extend the taskmodel incrementally and further develop the retrieval mechanism.

The scientific outcomes of this research are meant to be of value for both legal informatics and for information retrieval in general. The view on legal reasoning in

argumentation theory opposes the deterministic paradigm in which legal problems are solved automatically by the application rules. If the open-textured nature of legal problem solving is recognised, legal reasoning cannot be automated. This does not mean that information technology cannot be used to support legal tasks. I hope to have shown that it is possible to model legal tasks while respecting the discretionary nature of legal reasoning. A model as the one presented here, can be used to build legal information systems giving information support and methodical guidance for legal tasks rather than automatically derived conclusions. This may show the way for new types of legal information systems.

I have presented a method to use legal domain knowledge to shield users of legal databases from the internal operation of the retrieval mechanism. A legal taskmodel can act as an intermediary to bridge the conceptual gap between the user and the legal database. The conceptual gap that users experience using text retrieval systems stems from the generic nature of the applied technology. Therefore generic extensions and adaptions of the basic retrieval mechanism do not really bridge the conceptual gap. Only through the inclusion of domain knowledge in information retrieval systems, the conceptual gap towards the user can really be bridged. As the use of generic text database technology is widespread and the user-problems that it brings about are commonly recognised, this approach may also prove to be useful in other domains.

References

Agosti, M., R. Colotti and G. Gradenigo (1991). A Two-Level Hypertext Retrieval Model for Legal Data, in: A. Bookstein, Y. Chiaramella, G. Salton and V.V. Raghaven (eds.), Proceedings of SIGIR '91 Conference on Research and Development in Information Retrieval, October 1991, pp. 316-325.

Berger, F.C. and T.W.C.Huibers (1995). A Framework Based on Situation Theory for Searching in a Thesaurus, in: *The New Review of Document and Text Management*. Vol. 1, 1995, pp. 253-276.

Bruza, P.D. (1993). Stratified Information Disclosure: A Synthesis between Hypermedia and Information Retrieval, Amsterdam: Thesis, 1993.

Conklin, J. (1987). Hypertext : An Introduction and Survey, in: *Computer*. New York Vol. 20, no.9 pp. 17-41.

Dick, J.P. (1991). Representation of Legal Text for Conceptual Retrieval, in: *Proceedings of the Third International Conference on Artificial Intelligence and Law*, Oxford 1991, New York: ACM, 1993, pp. 244-253.

Eemeren, F.H. van, and R. Grootendorst (1992). *Argumentation, Communication, and Fallacies. A Pragma-Dialectical Perspective*. Hillsdale, NJ: Lawrence Erlbaum, 1992.

Hage, J.C., R. Leenes and A.R.Lodder (1994). Hard cases: a Procedural Approach, in: *Artificial Intelligence and Law*, Vol. 2, pp. 113-167.

Hair, D.C. (1991). LEGALESE: A Legal Argumentation Tool, in: *SIGCHI Bulletin*. Vol. 23, no. 1, pp. 71-74.

MacCormick, N. (1992). Legal Deduction, Legal Predicates and Expert Systems, in: *International Journal for the Semiotics of Law*. Vol. V, no.14. pp. 181-202.

Matthijssen, L.J. (1995). An Intelligent Interface for Legal Databases. in: *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, May 21-24 1995, College Park MD USA, New York: ACM, 1995, pp. 71-80.

Matthijssen L.J. (1996). Information Retrieval for Legal Argumentation Tasks. in: *Proceedings of the Fifth National/First European Conference on Law, Computers and Artificial Intelligence*, University of Exeter, April 15-16 1996, Exeter: E.U.C.L.I.D., 1996, p. 117-130.

Perelman, Ch. and L. Olbrechts-Tyteca (1969). *The New Rhetoric: a Treatise on Argumentation*, Notre Dame: University of Notre Dame Press, 1969. Translation of Traité de l'argumentation; la nouvelle rhétorique.

Prakken, H. (1993). *Logical Tools for Modelling Legal Argument*, Amsterdam: Free University of Amsterdam, 1993.

Royakkers, L. (1996). *Representing Legal Rules in Deontic Logic*, Tilburg: Tilburg University, 1996.

Salton, G. (1989). *Automatic Text Processing*, Reading, Mass: Addison-Wesley, 1989.

Toulmin, S.E. (1958). *The Uses of Argument*, Cambridge: Cambridge University Press, 1958.

Toulmin, S.E. (1992). Logic, Rhetoric & Reason: Redressing the Balance, in: F.H. van Eemeren et al. (eds.), *Argumentation Illuminated*, Amsterdam: International Centre for the Study of Argumentation (SICSAT), 1992, pp. 3-11.

Voermans, W. and E. Verharen (1993). LEDA: A Semi-Intelligent Legislative Drafting-Support System, in: J.S. Svensson, J.G.J. Wassink and B. Van Buggenhout (eds.), *Legal Knowledge Based Systems*, Proceedings 6th International Conference JURIX '93, Lelystad: Koninklijke Vermande, 1993, pp. 81-94.

Voermans, W. (1995). Modelling the Draughtman's Craft: the LEDA-project, in: *Proceedings of the Workshop Legimatics and Informatics for Legal Drafting*, Florence 25 November 1994, Florence 1995.

Zeleznikow, J. and A. Stranieri (1995). The Split-Up system: Integrating neural networks and rule-based reasoning in the legal domain. in: *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, May 21-24 1995, College Park MD USA, New York: ACM, 1995, pp. 185-194.