# BOKS: A Rule-based System in Support of the Dutch Building Materials Regulations

Pieter Spronck and Klaas Schilstra

TNO Building & Construction Research,
Knowledge Based Systems Department,
Rijswijk, The Netherlands
P.Spronck@cs.unimaas.nl
K.Schilstra@bouw.tno.nl

**Abstract.** BOKS is a knowledge based system which implements the "Bouwstoffenbesluit" (Bsb), the Dutch regulations on the use of building materials. Since the Bsb is a complex set of rules, which many people need to adhere to, a knowledge based system was seen as an ideal way to allow people to check whether they are using building materials according to the law. BOKS was implemented as a first generation rule-based system using decision tables. This paper discusses BOKS, the reasons for its success as a knowledge based system, its implementation and how that implementation could be improved upon, and makes a comparison between BOKS and related systems.

## 1 Building Materials Regulations

At the end of the twentieth century and the start of the twenty-first, concern for the environment in industrial countries becomes more and more important and people are seeking ways to deal with the problems of industrial waste. In this respect, the Dutch government actively promotes reuse of building materials, as long as this forms no threat to the health of the environment. With the 1995 publication of the "Bouwstoffenbesluit" (a governmental decision on building materials; its name is abbreviated to "Bsb"), the government strives to protect soil and water, while at the same time stimulating the reuse of secondary building materials. Since January 1, 1999, the Bsb has become a set of official regulations, and every building contractor is legally bound to conform to them [1].

The Bsb regulations are applicable in situations where primary and secondary building materials are transported to a building site to be used there. If there is no official quality assurance of the material to be used, a specified number of samples of the material must be taken to be examined in a laboratory, to determine the quantities and emissions of the different components in the material. Based on these laboratory findings, in combination with the exact purpose the contractor has for the material, the Bsb decides whether the material can be used, and if so, under which conditions.

The main part of the Bsb is concerned with the determination of the category of the building material. The different categories are:

- *Clean soil*: If the material falls in this category, it can be used without restrictions.
- *Category 1*: Category 1 materials can be used without restrictions in the situation the building contractor has specified. One of the consequences is, for instance, that if the contractor has specified that he will use the material in a layer with a specific height, he can't add to that height without running the risk of the material falling in another category.
- *Category 2*: Category 2 materials can only be used if they are isolated from the environment, and even then only in the particular situation the contractor specified.
- *Unusable*: Most building materials cannot be used at all if they don't belong to one of the preceding categories.

Besides these, there are two exemptions created for special building materials which the contractor is still allowed to use, even if they fall in the category "unusable" based on the laboratory findings.

## 2 Knowledge of the Bsb

The government publication of the Bsb consists of 300 pages of text [2]. This text is difficult to use, not only because of all the legalese, but also because there are numerous tables, formulas, references, schemas, footnotes and exceptions, which all have to be taken into account, and which are found distributed throughout the documents, often without references at the relevant points. Practice has shown that even Bsb experts find that it is difficult to use the Bsb correctly, because it easy to forget to take an exception or a footnote into account. On the other hand, once the Bsb has been applied to a particular situation, the results of the analysis are easy to understand.

To determine the category of a material, the Bsb expert just needs the laboratory findings and a description of the application situation. The answer needed from the expert is which of the six possible categories the material belongs to. The "input", that is, the laboratory supplied list and the situation description, can easily be understood and supplied by a layman. The "output", the category of the material, can be understood and used by a layman. The procedure to get from "input" to "output" is complex, however, even for experts, and even though the procedure is uniquely determined in the regulations.

So, while the knowledge laid down in the Bsb is well-defined, it is difficult to apply it in practice. Still, every building contractor is required to adhere to the Bsb. As early as in 1995, the Ministerial Department responsible for the development of the Bsb recognised the difficulties in making people use these regulations, unless they were made more accessible. The Knowledge Based Systems department of TNO Building & Construction Research was asked to determine if the Bsb could be implemented in a knowledge based system. TNO built a prototype, which resulted in 1998 in a project to implement a complete system. The first version was released early in 1999 under the name "BOKS" ("Bouwstoffenbesluit Ondersteunend Kennisgebaseerd Systeem"), at the same time the Bsb became a set of official regulations. The third version, implementing the latest changes to the regulations, was released in August 2000. Currently BOKS is successfully being used by about 3000 people, which amounts to the

majority of the potential user base. Even the few human experts on the Bsb use BOKS, since they agree it gives more accurate evaluations than they do themselves. Reviews of BOKS in magazines and newspapers have been generally favourable.

## 3 BOKS

A knowledge based system is a computer program which contains expert knowledge in a certain area, and which makes this knowledge available to the user. BOKS is a knowledge based system, which contains knowledge on the Bsb, or rather, knowledge of how an expert uses the Bsb. BOKS can, just like an expert, ask the user questions about his particular situation (like "what is the name of the material?", "where is it used?" and "what are the emission values the laboratory reports?"), and after the consultation provide the user with the same conclusions an expert would reach.

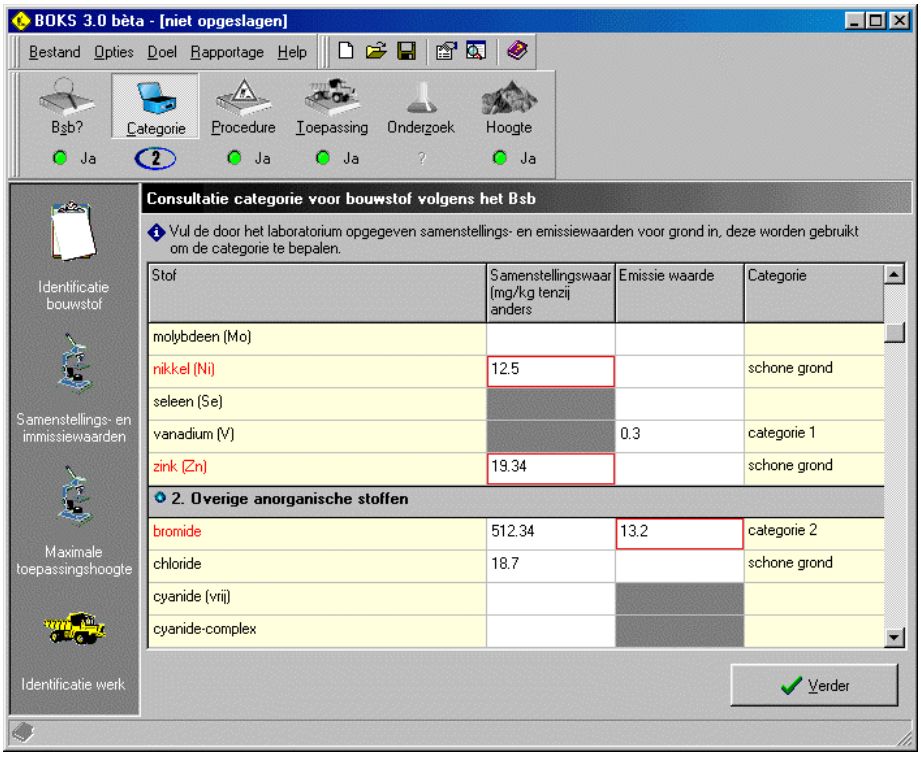



**Fig. 1.** BOKS shows the results of the laboratory research in a spreadsheet-like list. The rightmost column shows for each component what the category of the building material would be if that particular component would be the only one which was used to determine the category. The resulting overall category can be deduced from these intermediate categories. This final result is also visible under the category goal button at the upper left side of the screen - in this case, the encircled number "2".

The nature of the questions BOKS asks is such that a layman can answer them. Sometimes a situation arises in which a more difficult question is asked, one which most users may be able to answer, but some of them might not. In that case, the user is allowed to answer "I don't know", and BOKS will try to find out the answer to the question by asking a few different, easier to answer questions. BOKS determines in an intelligent way which questions it will ask. It contains about 150 different questions, but usually only about 30 of those are actually posed to the user during a consultation. The BOKS inference determines at each moment which question needs to be answered next.

As is usually the case with expert systems, BOKS can't discuss questions which aren't explicitly built into the system. The current release of BOKS can only determine the answer to six goal questions:

- Is the Bsb applicable to the user's situation?
- Which is the category the building material belongs to?
- What procedures must the user follow when he uses the material?
- What requirements are posed on the application of the material?
- How should a sample of the material be taken and examined?
- How high is the user allowed to stack the material?

These six goal questions (or rather, the first five goal questions) are sufficient to allow laymen to make sure they adhere to the Bsb.

## 4 Advantages of the Use of BOKS

The last goal question mentioned above (the determination of the maximum height the user is allowed to stack the material) is a nice example of a service BOKS provides, which can't be offered as easily by a human expert. Often, the category of a building material is limited by the height of the layer of the material which is used. Usually the user can stack the material even higher without changing the category, or have the material fall into a better category if he would just stack it a little lower. However, the formulas in the Bsb use the height of the material as input, and these formulas can't be inverted mathematically. Therefore it is not possible to directly calculate the maximum height of a material for a certain category. BOKS solves this problem by simply calculating the category for a few thousands of possible heights, thereby deducing at what levels the category changes. For a human expert this is far too much work, while BOKS performs this duty in the blink of an eye.

This is just one of the benefits the knowledge based system BOKS offers as opposed to a human expert. Some others are:

- Except for possible implementing errors in the knowledge base, BOKS is infallible and complete. A human expert may forget to take an exception to a rule into account, but BOKS won't.
- BOKS can be replicated. A human expert can only work in one place, while BOKS can be used in many places at the same time.

- BOKS can be used at any time, while a human expert is only available for part of the time.
- BOKS' results are consistent and can be reproduced. While two human experts may have a different view of a particular situation, two copies of BOKS will always provide the same answer.
- BOKS has built-in reporting facilities. For a human expert, the writing of reports is a tedious and time-consuming activity. BOKS generates its reports automatically.
- It is easy to keep the knowledge in BOKS up-to-date. Because the knowledge is implemented in decision tables with the use of a special tool (which will be discussed in the next paragraph), when regulations change it isn't difficult to locate in the knowledge base the rules which need to be changed.
- It is relatively easy to get users to adhere to new versions of the regulations. If the Bsb changes, which happens regularly, the BOKS knowledge base must be updated once and then be redistributed to the users. This is far easier and cheaper to do than retraining all the human experts.

The many benefits BOKS offers as opposed to a human expert (most of which, incidentally, are offered by knowledge based systems in general [3]) seem to suggest that a knowledge based system is to be preferred over a human expert. In the case of BOKS, this is probably true. The reason is that the knowledge domain BOKS covers, a set of regulations, is because of its nature complete, consistent, and well-defined, and is therefore particularly suited to be implemented in a rule-based system. Since a large number of Bsb experts are needed in practice, BOKS is an ideal alternative for the employment of human experts. Even if BOKS doesn't cover the Bsb completely, the questions it does answer (which are all the questions needed for people to make sure they adhere to the regulations), it answers with 100% accuracy. Most knowledge bases only cover part of the knowledge domain they implement, and aren't able to answer a question in all circumstances. As such, those knowledge bases can only be used in support of a human expert (which is, of course, also valuable). BOKS is an exception in this respect.

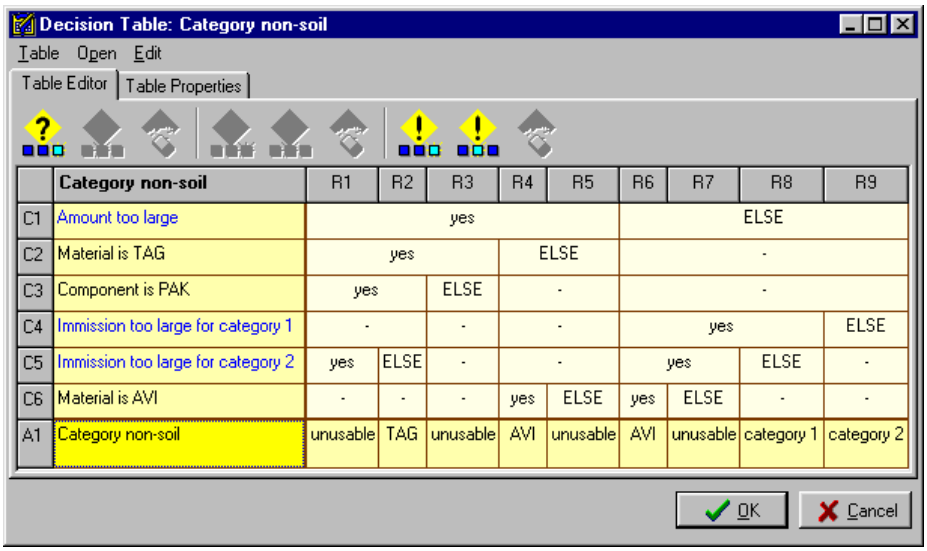
## 5 Implementation of BOKS

It is common practice with knowledge based systems to place the actual knowledge outside the program itself, in a knowledge base. The main reason for the existence of this separate knowledge base is that it should be easy to maintain the knowledge. Not only can such a knowledge base be changed without changing the actual program, it can also be maintained using a special tool, which allows the user to view and edit the knowledge in a user-friendly way, so even some non-programmers can do it.

A common method to store knowledge in a knowledge base is in the form of rules. Usually rules are "IF...THEN..." statements, like "IF the building material is used on the inside of a building THEN the Bsb is not applicable". There are alternative ways to implement knowledge (like in the form of "cases" or in a neural network), but a knowledge domain consisting of regulations is particularly suited to be implemented

in a rule-based system, and BOKS does just that. The BOKS knowledge base is built and maintained using the "Knowledge Base Editor" (KBE). This is a tool developed by TNO which maintains knowledge bases consisting of "decision tables" [4]. A decision table is a visual representation of a piece of rule-based reasoning (Fig. 2). BOKS contains about 180 of these tables.

The BOKS knowledge base was built by a knowledge engineer (KE) who worked in close collaboration with Bsb experts. The KE used interviews to question the experts on the use of the Bsb and implemented the results of those interviews in decision tables. This often led to new questions, which were again posed to the experts. This is a regular "propose and revise" approach to knowledge acquisition. Because of the complexity of the Bsb, the KE couldn't directly use the regulations themselves to build




| | Category non-soil | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 |
|---|---|---|---|---|---|---|---|---|---|---|
| C1 | Amount too large | yes | | | | | ELSE | | | |
| C2 | Material is TAG | yes | | | ELSE | | - | | | |
| C3 | Component is PAK | yes | | ELSE | - | | - | | | |
| C4 | Immission too large for category 1 | - | | - | - | | yes | | | ELSE |
| C5 | Immission too large for category 2 | yes | ELSE | - | - | | yes | | ELSE | - |
| C6 | Material is AVI | - | - | - | yes | ELSE | yes | ELSE | - | - |
| A1 | Category non-soil | unusable | TAG | unusable | AVI | unusable | AVI | unusable | category 1 | category 2 |

**Fig. 2.** This is a decision table which determines the category a non-soil building material belongs to according to one particular component. The left column (except for the bottom row) contains "conditions". To the right of the conditions are the "condition alternatives". Conditions have a value, which is determined either by asking the user for it, or by reading the value from a database, or by executing another decision table. That value is then checked against the alternatives, which determines the path that will be taken to get to the answer. The first condition is "Amount too large" (C1). If the Bsb sets an upper limit to the quantity of the component in the material and that limit is exceeded, this condition contains "yes". In that case, the left side of the alternatives (columns R1-R5) is used to continue the evaluation process, and otherwise the right side (columns R6-R9) is used. The next condition is "Material is TAG" (C2). If the quantity limit wasn't exceeded, this condition is skipped, as the dash in row C2, columns R6-R9 shows. The same holds for the condition after that (C3). By following the condition alternatives belonging to the value of the conditions, the evaluation process ends up in one of the cells on the bottom row. These cells contain the "action alternatives", and the value of the cell the evaluation ends in is given to the "action", which is contained in the lower left cell, "Category non-soil" (A1). For example, if the quantity limit is not exceeded, and the immision for category 1 isn't exceeded, the component category is "category 1".

the knowledge base, though he could use parts of it, after the experts had indicated which sections were of importance for the implementation.

The KBE allows the consultation of partially finished knowledge bases, even of individual decision tables, and the KE and the experts used this to test the implemented knowledge. The KE often chose to implement a small piece of knowledge, for instance a calculation or an exception specified in a footnote, in one separate decision table. This piece of knowledge could then be tested separately from the whole knowledge base. The decision tables themselves were also discussed with the experts. Because of the way decision tables make an inference visible, the experts were able to review the knowledge implemented in the knowledge base, and could easily determine whether the knowledge was correct and complete.

Besides a knowledge base, BOKS contains a database of building materials, for which many of the answers to questions the knowledge base might ask are already known (though BOKS is not restricted to these materials). Bsb experts can easily add to this database. The BOKS reports are also stored in a separate file, which has an XML structure, and those reports can therefore also be maintained by the Bsb experts. Last but not least, BOKS contains the complete text of the Bsb, and the reports hyperlink to the relevant paragraphs in this text. The BOKS program itself is implemented using the Borland development environment "Delphi".

The time needed for the implementation of the first version of BOKS was about 1000 man hours, not counting the work which was spent on a prototype finished in 1996. About 25% of the time went to the building of the knowledge base, 60% to the building of the application and the reports, and the remainder went to testing and administrative tasks. This excludes the time experts spent on the development of the system, but practice indicates that this must have been about half the man hours spent by the developers.

Because of BOKS' user-friendliness and the fact that BOKS only asks simple questions, people often suppose BOKS is no more than a friendly kind of spreadsheet. While the programmers are quite pleased with that, since it shows the design of BOKS is successful, that impression is not correct. Because of the complexity of the Bsb, BOKS could only be implemented in a spreadsheet if it was augmented with a lot of program code, and even then, practice has shown (because this has actually been tried), it would only be able to give partial answers to questions. On top of that, in a spreadsheet "solution" most of the knowledge would reside in program code instead of in a knowledge base, making it virtually impossible for Bsb experts to check the knowledge, let alone maintain it.

## 6 Improving BOKS

There are several improvements which would benefit BOKS. The first and most obvious is that the user should be able to access BOKS over the Internet. Changes to the regulations, implemented in the knowledge base, would then be immediately available to users, while currently they have to download and install a new stand-alone program. The reason that BOKS isn't an Internet based system is mainly one of budget: recent

developments to the KBE allow the deployment of knowledge bases as easily over the Internet as as a stand-alone application [5], but these techniques weren't available at the time BOKS was developed, and the budget to enhance BOKS, like with this new functionality, is very limited and goes mostly to updates to the knowledge base.

The second improvement would be to use an object oriented knowledge base for BOKS instead of a regular rule-based knowledge base. While the current knowledge base is certainly adequate for BOKS, an object oriented knowledge base would improve maintainability. At the time BOKS was developed, however, there was no tool available which would combine an object oriented knowledge model with the visual representation of knowledge (in decision tables, for instance) and the easy integration with a regular development tool. There has been demand for such a tool for some years now [6], and at TNO one is currently in development. It is called "Intelligent Objects" (IO) and will become available in 2001 [7]. If at that time a redesign of BOKS is desired, it will probably be done with this tool.

## 7 Comparison with Related Technologies

As a regular rule-based system BOKS has many of the features in common with classic systems such as MYCIN [8] and XCON [9], which basically belong to the category of first generation knowledge systems. As the term implies, second generation systems also exist and are considered to be more mature. These second generation systems can be seen as a response to the knowledge acquisition bottleneck, where the difficulty of transferring knowledge from the expert to a computer representation was seen to fail. This failure was commonly attributed to communication problems and a mismatch between human and computer representations [10]. A prime example of a second generation approach is CommonKADS, a knowledge engineering methodology that employs deep models to allow knowledge acquisition and maintenance to be based on the design of an operational model of the domain. Modelling the knowledge used explicitly is seen as an important aspect to construct comprehensible, controllable, maintainable systems [11].

A problem that both first and second generation systems seem to fall victim to is the gap between prototype and industrial strength knowledge systems [12]. The effort necessary to fit a knowledge system with a dedicated user interface and develop auxiliary facilities can constitute the majority of the development effort of a knowledge system, caused in part by inadequacies in the tools used. This problem negatively affects the acceptance of systems and can even lead to knowledge systems never reaching their intended users.

The focus of the effort to develop BOKS employed a different route from the one the model based approaches advise. An iterative and incremental development of the knowledge was used, with participation of the expert in the modelling process, and with a focus on developing an industrial strength system for a large group of users. This is comparable to the evolutionary and business driven approach found in regular software engineering techniques, like DSDM [13] and Rational Unified Process [14]. An important part of the development process was to design how the system would be

visualised and used, and the experts and users were also involved in decisions regarding those aspects. The fact that components were available to integrate KBE knowledge bases in a regular application, allowed great flexibility on the part of the developers to make an application to the user's liking.

Therefore, relating the development of BOKS to current directions of research, BOKS has shown that a successful system, employed by many users, can be created using a technology that has more in common with first generation than with second generation systems. This does not entail that approaches such as CommonKADS are wrong. It does, however, raise questions regarding the assumptions on why a specific approach to knowledge system development truly works well.

## 8 Conclusion

BOKS is an advanced, first generation knowledge based system which combines a user-friendly interface with complex knowledge on the Dutch "Bouwstoffenbesluit" (Bsb), laid down in a rule-based knowledge base using decision tables. BOKS allows laymen in the area of the Bsb to adhere to the regulations it contains, without the need of them obtaining deep understanding of the actual Bsb regulations. The system has been developed using an iterative and incremental approach to the development of the knowledge and the application as a whole, with participation of the experts in the modelling process, and with a focus on developing an industrial strength system for a large group of users. BOKS has shown in practice to be a successful and easily-used implementation of the Bsb, with a majority of the intended user base actually employing the system.

## References


1. Aalbers, Th. G. *et al*.: Bouwstoffen nader bekeken. Eburon, Delft, (1998).
2. Ministerie van VROM: Bouwstoffenbesluit, parts 1, 2 and 3 (1998).
3. Turban, E.: Decision Support and Expert Systems: Management Support Systems, 2nd ed. MacMillan, New York (1990).
4. Montalbano, M.:Decision Tables. Science Research Associates, Inc, Palo Alto, California (1973).
5. Spronck, P. and S. Peeters, S.: iDoc: Rule-based Knowledge Bases on the Internet. White paper on the website of the authors' employer: www.bouw.tno.nl/bkr (2000).
6. Luger, G.F. and Stubblefield, W.A.: Artificial Intelligence, Structures and Strategies for Complex Problem Solving, 3rd edn. Addison Wesley Longman, Inc., Reading, USA (1997).
7. Spronck, P.H.M. and Schilstra, K.O.: Objectoriëntatie bij kennismodellering. In: Informatie, July/August 1999. Ten Hagen Stam Uitgevers (1999) 21-26. An English version of this paper, titled Object Knowledge Modelling, is found on the website of the authors' employer: www.bouw.tno.nl/bkr.
8. Shortcliffe, E.H.: Computer-based medical consultations: MYCIN. Elsevier, New York (1976).
9. Bachant, J. and McDermott, J.: R1 revisited: Four years in the trenches. AI Magazine 5(3)

(1999) 21-32.
10. Buchanon, B., Barstow, D., Bechtel, R. *et al*.: Constructing an Expert System. In: Hayes-Roth, F., Waterman, D. and Lenat, D. (eds.): Building Expert Systems. Addison-Wesley, Reading MA (1983).
11. Schreiber, G., Akkermans, H., Anjewierden, A. *et al*.: Knowledge Engineering and Management. MIT Press, Cambridge (1999).
12. Crofts, A.E., Ciesielski, V.B., Molesworth, M., Smith, T.J., and Lee, R.Y.: Bridging the Gap Between Prototype and Commercial Expert Systems - A Case Study. In: Quinlan, J.R. (ed.): Applications of Expert Systems. Addison-Wesley, Reading MA (1989) 93-105.
13. Stapleton, J.: DSDM: Dynamic Systems Development Method. Addison-Wesley, Reading MA (1997).
14. Jacobson, I., Booch, G. and Rumbaugh, J.: The Unified Software Development Process. Addison-Wesley, Reading MA (1998).

## Software

BOKS is freely available from the Dutch Ministry of VROM. It can be downloaded from www.minvenw.nl/rws/dww/boks. A preview version of the KBE, which is used to implement the BOKS knowledge base, is available from www.bouw.tno.nl/bkr.