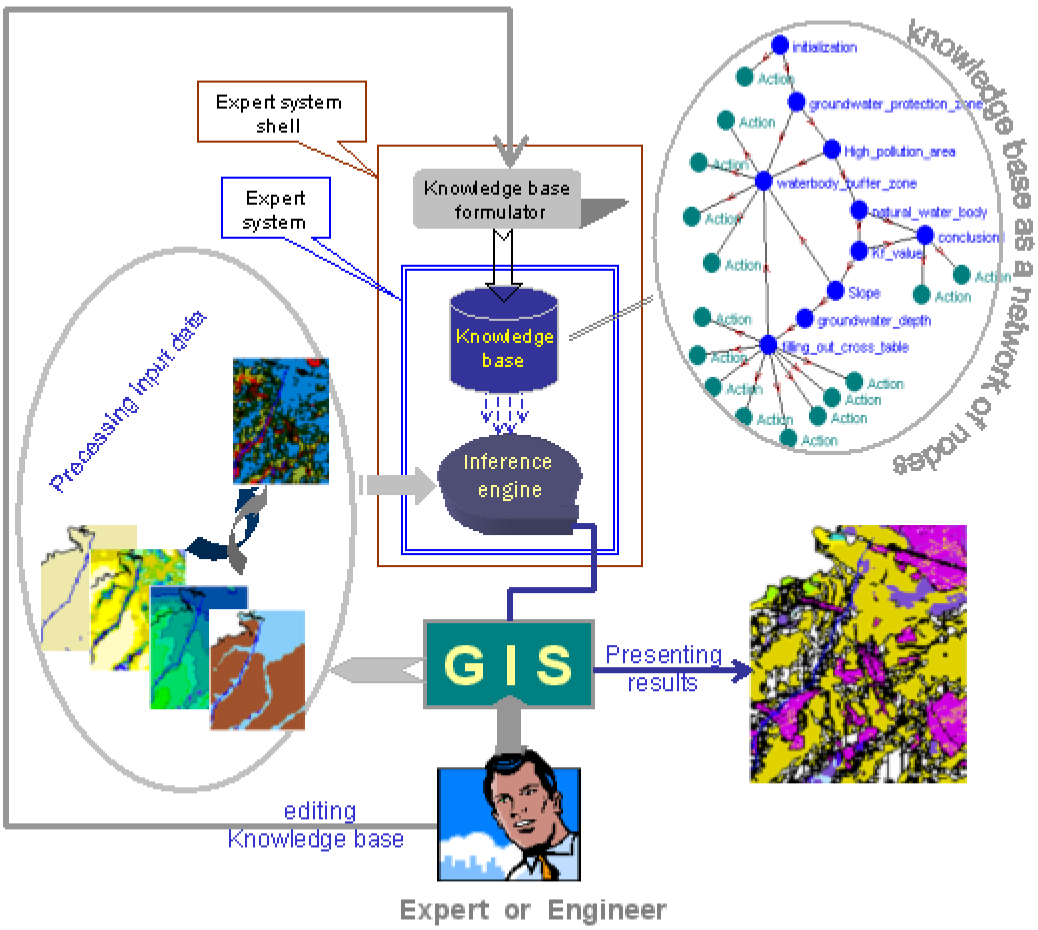
Development of a Transparent Knowledge-Based Spatial Decision Support System for Decentralised Stormwater Management Planning

Case study: Selection of On-site Stormwater Management Measures
for Urban Catchments: Chemnitz and Emscher Region, Germany



Development of a Transparent Knowledge-Based Spatial Decision Support System for Decentralised Stormwater Management Planning

Case study: Selection of On-site Stormwater Management Measures
for Urban Catchments: Chemnitz and Emscher Region, Germany

Von der Fakultät
Bauingenieurwesen und Geodäsie
der Universität Hannover
zur Erlangung des Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN

Dr.-Ing.
genehmigte Dissertation von

M.Sc. Zhengyue Jin

geboren am 18 Oktober 1963
in Anhui, VR.China

Dezember 2005

Referent: Prof. Dr.-Ing. Friedhelm Sieker

Korreferent: Prof. Dr.-Ing. Theo G. Schmitt

Tag der Promotion: 14.Dezember 2005

Schlagworte

Dezentrale Regenwasserbewirtschaftung, Expertensystem, Entscheidungshilfesystem

Zusammenfassung

Die dezentrale Regenwasserbewirtschaftung ist bedeutsam und verbreitet in der Planung der Stadtentwässerung, denn sie hat im Vergleich zu herkömmlichen Ableitungsverfahren viele Vorteile. Aber die Klärung der Frage, ob ein privates Gründstück oder eine öffentliche Fläche vom Kanalnetz abgekoppelt bzw. ob die Regenabflüsse vor Ort versickert werden können, hängt von der Bewertung einer Vielzahl von Parametern ab. Eine systematische bzw. computergestützte Bewertung ist derzeit nicht möglich. Dies wäre aber sinnvoll, wenn man an die Planung von dezentralen Regenwasserbewirtschaftungsmaßnahmen für große Stadteinzugsgebiete denkt.

Die vorliegende Arbeit lässt sich hauptsächlich in zwei Teile gliedern:

Im ersten Teil wurde ein transparentes, wissensbasiertes, und räumliches(GIS Daten als Grundlage) Entscheidungshilfesystem als ein Instrument zur Automatisierung der Bewertung der dezentralen Regenwasserbewirtschaftungsmaßnahmen konzipiert bzw. entwickelt. Dieses Instrument integriert ein wissensbasiertes Expertensystem (ES), weil die Lösungen der o. a. Probleme dynamisches, empirisches und fachliches Wissen erfordert.

Das in dieser Arbeit entwickelte Expertensystem hat aber eine Besonderheit gegenüber einem normalen Expertensystem. Da dieses Expertensystem als ein Entscheidungshilfesystem genutzt werden soll, muss dem Anwender die Wissensbasis transparent vorliegen und von ihm auch modifizierbar sein. Zu diesem Zweck wurde ein eigenes GIS-integriertes Expertensystem-Werkzeug entwickelt, welches die o. a. Anforderungen erfüllt. Die Lösung besteht aus „GIS + ES tool + Wissensbasis“. Die Wissensbasis, in diesen Fall die Bewertungsregeln, wird aber jedoch erst im den zweite Teil dieser Arbeit entwickelt.

Der zweite Teil der vorliegenden Arbeit untersucht hauptsächlich zwei praktische Anwendungen: Möglichkeiten von dezentralen Regenwasserbewirtschaftungsmaßnahmen für die Stadt Chemnitz und für das Einzugsgebiet der Emscher. Zu dem o.a Instrument (GIS + ES tool), wird für jedes betrachtete Gebiet eine projektspezifische Wissensbasis aufgebaut. Die Vielzahl von räumlichen Daten von Einflussfaktoren für die Einzugsgebiete erfordert ein strukturiertes Vorgehen bei der

Regelerstellung. Zuerst wird jeder Einflussfaktor hinsichtlich seiner vorhandenen Datengrundlage in Bezug auf die dezentrale Regenwasserbewirtschaftung klassifiziert (z.B. gibt der Durchlässigkeitsbeiwert Auskunft darüber, ob eine vollständige Versickerung möglich ist oder nicht.). Die erstellten Regeln ergeben ein Bewertungsfließdiagramm, das je nach Gebiet und Datengrundlage variiert. Die Regeln sollten möglichst flexibel angelegt sein, so dass Veränderungen leicht integrierbar sind. Als Beispiel sei das Emscherprojekt erwähnt. Das Bewertungsfließdiagramm ist als Netzwerkstruktur (nicht als Baumstruktur) angelegt. Um eine sehr große Vielzahl an Kombinationen zu vermeiden, wurde der Regelstruktur ein Kreuztabellenverfahren zu Grunde gelegt, welches die Anzahl an Lösungen festlegt. Das wissensbasierte Bewertungsfließdiagramm ist im Format des Expertensystems abgespeichert und kann beliebig oft verwendet werden.

Die Ergebnisse der Durchführung der beiden wissensbasierten räumlichen Entscheidungshilfesysteme für Chemnitz und für das Einzugsgebiet der Emscher wird im GIS graphisch präsentiert. Als wesentliches Ergebnis entsteht eine Karte mit der räumlichen Verteilung von möglichen dezentralen Regenwasserbewirtschaf-tungsmaßnahmen. Das Expertensystem liefert in der Datenbank ebenso die Information, auf Grund welcher Einflussfaktoren es zu der entsprechenden Bewertung kam (Nachvollziehbarkeit der Entscheidung ist gegeben). Ebenso können Bemerkungen zu Teilgebieten auf fehlende Daten hinweisen.

Die entwickelten transparenten wisssensbasierten Entscheidungshilfesysteme geben dem Ingenieur ein flexibles und allgemeines Instrument zur Planung von dezentralen Regenwasserbewirtschaftungsmaßnahmen.

Keywords

On-site stormwater management, expert system, spatial decision support system

Summary

On-site rainwater management is becoming a significant and prevalent supplement method in urban drainage planning and has many advantages compared to conventional discharge processes. The question of whether a piece of private property or public space can be disconnected from the sewer system and infiltrated on-site, however, depends on the evaluation of multiple parameters. A systematic and automated evaluation with the computer is not possible at present. This would be rather sensible, in particular, when one considers planning on-site stormwater management measures for large urban catchments.

The present study is divided mainly into two parts:

In the first part, a transparent, knowledge-based and spatial decision support system was conceptualised and developed as an instrument to automate the evaluation of on-site stormwater management measures. This instrument integrates a knowledge-based system or an expert system (ES) because the solutions to the above-mentioned problems require dynamic, experiential and professional knowledge.

The expert system developed in this task, however, has a distinctive feature compared to a normal expert system. Because this expert system is supposed to be used as a decision support system, the knowledge base must be transparently available to the user, as well as modifiable by him. For this purpose, a GIS-integrated expert system tool was developed that fulfills the requirements mentioned above. The overall solution consists of “GIS + ES tool + Knowledge Base.” The knowledge base, in this case the evaluation rules, is only developed in the second part of this task, however.

The second part of the present study examines mainly two practical applications: possibilities for on-site stormwater management measures for the city of Chemnitz and the Emscher catchment. For the above-mentioned instrument (GIS + Es tool + Knowledge Base), a project-specific knowledge base is constructed for each considered area. The multitude of spatial data about influencing factors for the catchments requires a structured procedure while formulating rules. First, each

influencing factor in consideration of its existing data is classified regarding on-site stormwater management; for example, whether the permeability coefficient provides information as to whether complete infiltration is possible or not. The compiled rules result in an evaluation diagram that could vary for different catchments or different data situations. The initial rules or diagram should be constructed as flexible as possible so that changes are easily integrable. For example, for the Emscher project, the evaluation diagram is established as a network structure, not as a tree structure. It applies a so-called cross-table method that avoids the enumeration of too many possible solutions. The evaluation diagram is further translated into and saved as a ES knowledge base in the format designed in this study and can be used and modified at any time.

The established two knowledge-based spatial decision support systems for Chemnitz and the Emscher catchment (i.e. two specific knowledge bases applied in the above-mentioned overall model respectively) are applied to the prepared GIS databases respectively and the results are presented graphically in GIS. For each evaluation, the fundamental result is a map of the spatial distribution of possible on-site stormwater management measures. In the database, the expert system also indicates information regarding upon which influencing factors an evaluation was made. Likewise, comments about missing data in certain sub-areas are pointed out too.

The developed transparent knowledge-based decision support systems give the engineer a flexible and general instrument for planning on-site stormwater management measures.

CONTENTS

1. Definition of problem and goal

1.1 Integrated planning of on-site stormwater management measures

1.1.1 Urban drainage problem

With the development of the economy, the migration of people from rural to industrialised areas, i.e. urbanisation, accelerates worldwide. As a result, large amounts of natural areas, such as meadows, forests, agriculture lands, etc. become sealed due to the construction of apartments, factories, institutions, sports and entertainment buildings or yards, roads, streets etc.. For example, nowadays in Germany, statistically about 122 ha of natural area are consumed for new apartments and streets daily, of which around 50 percent becomes impervious(Sieker, *et al*, 2002). In fact, the urbanisation process is occurring even more rapidly in developing countries like China (Koop, 2001), India, etc.. An excerpt from “The Key Facts About Cities, Issues for the Urban Millennium, United Nations Environment Programme” gives a general idea of the global urbanisation process.

> In 1950 less than one person in three lived in a town or city. Today nearly half the world’s population is urban. By 2030 the proportion will be more than 60 per cent. Virtually all population growth in the next quarter-century will be in urban areas in the less developed countries.

Urbanisation, on one hand as a symbol of economical development, brings out on the other hand many negative effects. One of the most direct problems is the overloading of the city sewer system, which is at present the prevailing urban water drainage system in the world. The overloading of an existing sewer system is due to more and more surface runoff converged from continuously newly connected impervious areas.

For a combined sewer system, that is, rainwater and wastewater drained together in one pipe system, the overloading means on one hand the hydraulic overload in different points inside the pipe system, on the other hand the overloading of the wastewater treatment plant at the end of the pipe system, or more overflow of the combined sewer directly into the receiving water.

For separate sewer systems, that is, rainwater drained in its own pipe system, the overload on one hand means the hydraulic overload at different points inside the pipe system, and on the other hand more rainwater without any treatment directly into the receiving water. So in both cases, urbanisation will cause hydraulic overloading inside the pipe system and a higher pollution load in the receiving water.

To solve the hydraulic overload problem in the city, the conventional method is the rehabilitation of the sewer system by enlarging relevant pipes. However, this actually worsens the second problem, i.e. receiving water pollution, because more and more rainwater or combined sewage will be directly drained into the receiving water. Although the combined sewer overflow can be reduced through the construction of central water retention basins, there is the same problem as for the existing sewer pipe system, i.e. the retention basin must be continuously enlarged, or new retention basins continuously constructed, with the continuous increase of inflow from the sewer system. This is furthermore a problem of available space to construct the retention basins and the high cost of construction for basins and pumping stations.

Aside from the water pollution problem, the continuous enlargement of sewer pipes could transfer the flood problem caused by urbanisation from inside the city to nearby streams or rivers. This is especially significant in some small river basins with high urbanisation. Another negative effect of draining rainwater totally through a pipe system is that the water balance in the drained area will worsen because of the reduction of evapotranspiration and groundwater recharge. The reduction of evapotranspiration affects the local climate to some extent, while the reduction of groundwater recharge endangers sustainable social development regarding water resources.

1.1.2 New concept on urban rainwater management

Considering the above-mentioned problems originating from urbanisation and the shortcomings of the conventional solution, a new concept of urban water management, decentralised rainwater management, which aims to reduce surface runoff to the existing sewer system during rain events, is being gradually put into practice. For example, Sydney, Australia, has a stormwater system that is completely separate from the sewer system by using on-site detention (OSD) measures (S Beecham et al, March, 2005). In the USA, many states have a stormwater best management practice (BMP) manual which includes, among other things, detailed instructions to the selection, design and construction of the most suitable on-site stormwater mangement measures (for example, in Chapter 9, New Jersey Stormwater Best Management Practices Manual). On-site stormwater management is becoming popular nowadays in many other countries as well. Figure 1 shows two photos of pratical examples of on-site stormwater management structures in

Auckland, New Zealand (On-site stormwater management manual, Auckland, New Zealand)

In Germany, which is among the first countries in urban hydrology research and practice, on-site stormwater management, especially on-site infiltration, has been extensively researched and widely put into practice. Researchers have developed many new concepts on on-site infiltration and accumulated a lot of theoretical and practical knowledge in this aspect. For example, Fig.2 demonstrates schematically the application of the combination of on-site infiltration and the traditional underground pipe drainage, which introduces the concept of incomplete infiltration.

Fig. 1 On-site stormwater management measures applied in Auckland, New Zealand (source: On-Site Stormwater Management Manual Auckland City, 2004)

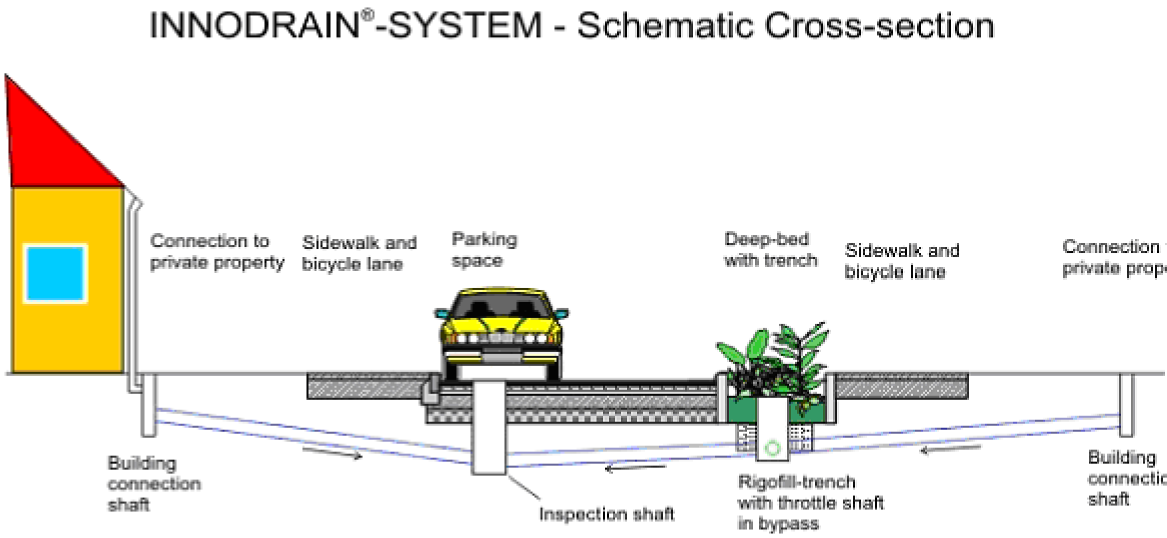


Fig. 2 Scheme of the cross section of an INNODRAIN-SYSTEM (Source:F. Sieker *et al*, 2004)

1.1.3 Integrated planning of on-site stormwater management measures in an entire urban stormwater drainage system

Among on-site stormwater management measures, rainwater utilisation is a relatively simple concept. Its application is rather limited to household water use or garden irrigation, etc.. The boundary conditions on the installation of a rainwater utilisation system depend mainly on the integration of the system (for example, a storage tank) with the local environment and the construction and operating costs.

A decentralised rainwater retention basin functions much like a central retention basin, but functions on a reduced scale and inside a drainage catchment. The effects of decentralised retention basins include: relieving the hydraulic overloading of the sewer system, delaying the concentration of runoff into the receiving water and reducing the combined overflow (when the system is the combined sewer system). Whether a decentralised retention basin should be constructed to relieve the overloading of sewer pipes mainly depends on the available natural area, the management of the retention basin, cost and construction issues.

On-site rainwater infiltration tries to make the best use of the storage capacity of local soil. The rainwater from a local impervious area is collected and drained into an infiltration structure. The collected water will either immediately infiltrate into the surrounding soil or infiltrate after temporary storage. In this way, constructed areas suitable for this type of treatment can be disconnected from the sewer system and drained on-site. Obviously, on-site rainwater infiltration is more often applicable than rainwater utilisation for its suitability to disconnecting more types of built-up areas and it is more flexibly installed than a decentralised retention basin considering its requirement for natural area and its integration into the surrounding environment. The foremost advantages which make infiltration measures desireable are:

- reducing inflow to the existing sewer system, hence reducing flooding inside the city or contributing to flood control in rivers in or near cities.
- contributing to the cleanliness of the rainwater by filtering rainwater through vegetated shallow ponds and/or active soil layers (see **5.1.1**).
- increasing groundwater recharge and to some extent also evapotranspiration. In this regard, on-site rainwater infiltration manages rainwater in a near-natural way.

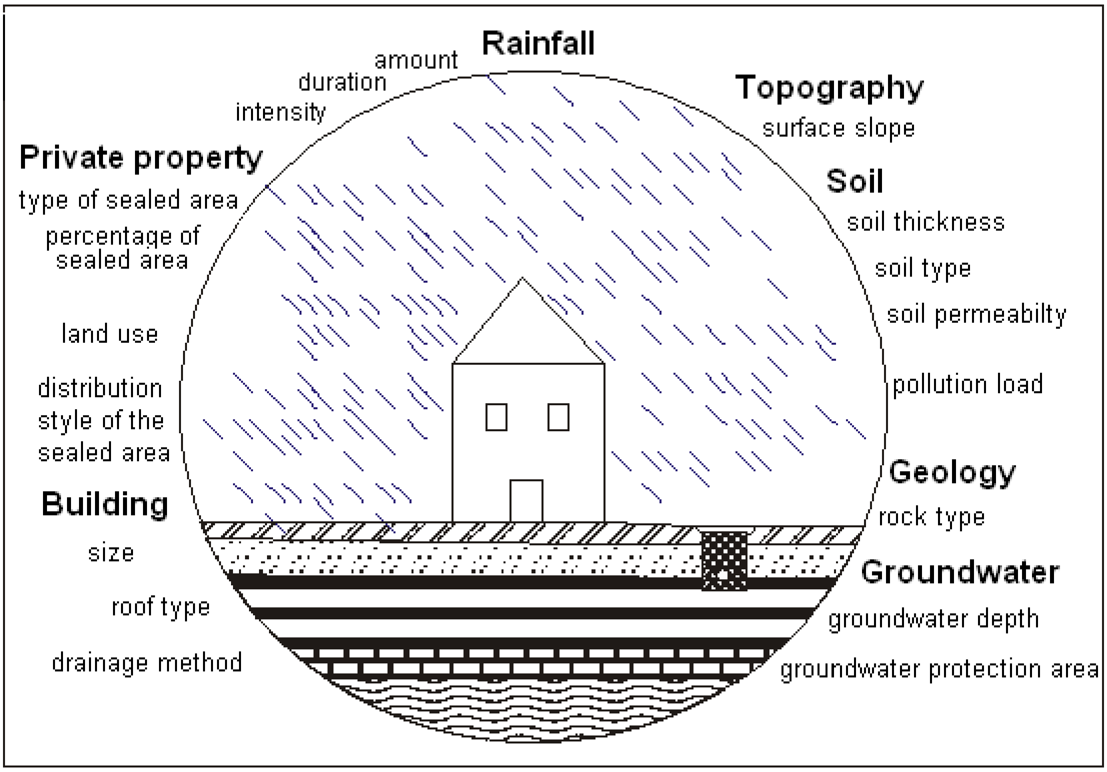


Fig. 3 Different influencing factors on the selection of on-site stormwater infiltration measures (source, S. Bandermann, 2002)

However, compared to the other two types of decentralised rainwater management, the application of on-site infiltration measures depends on not only factors related to construction and cost, but also influencing factors such as soil type, soil permeability, groundwater depth, stormwater pollution load, etc. (Fig. 3). The decision on whether on-site stormwater infiltration is suitable for a given area, and if so, which type of infiltration measure is the optimal one for that area involves relatively complicated, multi-disciplinary and rather experienced judgement. The complexity of this decision or selection is especially remarkable if one considers that the complicated decision must be made individually for numerous small areas (such as a single lot of a villa or an apartment) during an integrated planning of the stormwater drainage for a whole urban catchment. Moreover, such decisions usually need to be repeated under the change of the planning constraints because planning normally requires the comparison of the hydrological and hydraulic effects of many different scenarios.

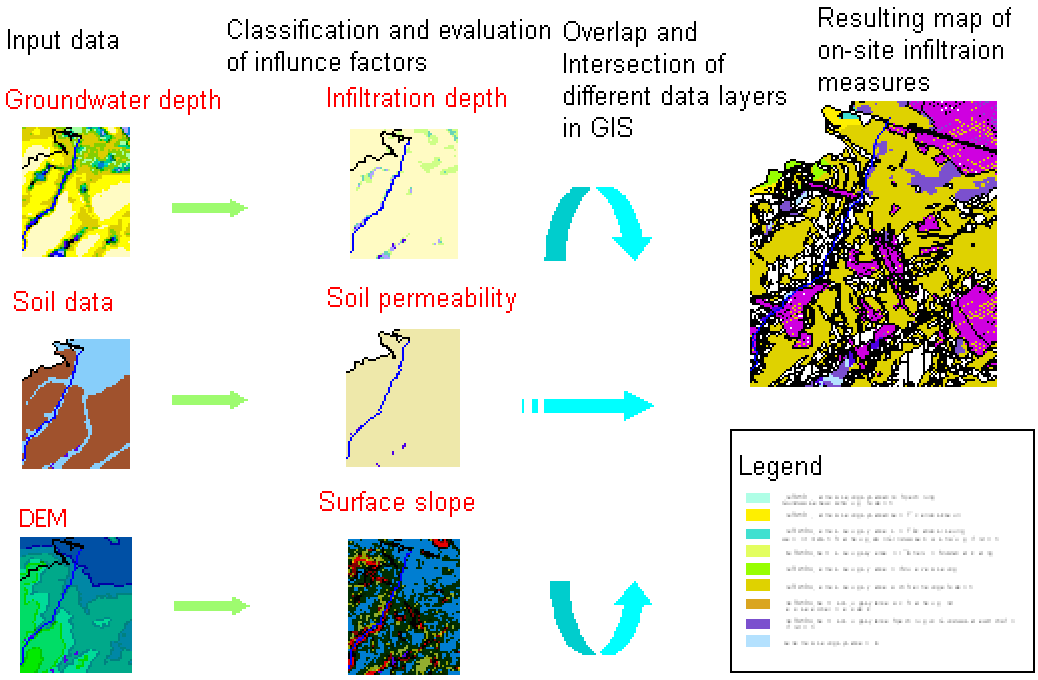


Fig. 4 A schematic demonstration of the procedure of planning on-site infiltration measures (source, F. Sieker, 2003)

A schematic presentation (Fig. 4) of the general procedure of planning on-site infiltration measures for an urban catchment demonstrates the complexity of the process. First, the different digital input data (GIS data), such as groundwater depth, soil permeabilty, etc. are processed and classified according to their individual boundary conditions on each of the stormwater infiltration measures. Some factors may not be directly available and hence must be derived from other original data; for example, the surface slope data is usually derived from digital elevation model (DEM). The classified spatial data are normally distributed in a different style and hence held in different layers of a GIS project file. Therefore, they must then be spatially overlapped and intersected with each other to obtain a new spatial distribution (i.e. a layer), in which data of all factors are included and consistent (that is, values of each factor are kept consistent in every object of the layer). Because of the spatial intersection of many layers, the number of objects in the resulting layer could be hundreds of thousands. This means there must be hundreds of thousands decisions to be made for one planning scenario. Moreover, each decision process is dynamic because the effects of some factors on the decision of the infiltration measure may depend on what other parameters have currently affected the decision. For example, when the groundwater depth is relatively high, then the on-site infiltration structures with underground storage facilities (such as a trench) are usually ruled out, and further evaluation of the peameability of deep soil layer is not necessary. Such dynamic unstructured analysis is difficult to carry out through macro operations in a database recordset or EXCEL sheets, which could be a way to deal with the massive decisions. Therefore, the selection of a suitable on-site infiltration measure needs to be manually analysed for one object after another. This is not really practicable.

1.2 Decentralised water management planning aimed at preventive flood control in river basins

Similar to on-site stormwater management in urban areas, the preventive decentralised measures on a river catchment scale aims to reduce as much on-site surface runoff in the whole river catchment as possible during rain events. One decentralised management concept is the increase of natural water infiltration in agricultural lands, forested area, etc.. This concept tries to make the best use of the water-retention capacity of the soil in the whole catchment and concerns every patch of land where the potential exists for storing rainwater during rain events. Compared to conventional river engineering, such as channelisation, this preventive decentralised measure on one hand mitigates potential floods to some extent, and on the other hand increases base flow as well as soil moisture after flooding. The special characteristics of this kind of preventive decentralised measure is that it makes a contribution to flood control without just shifting the problem from the upper reaches to the lower reaches of a river basin, as is the case with some conventional river engineering measures like chanellisation.

However, planning the decentralised measures is based on the appropriate evaluation of soil water retention potentials in widely diverse situations in the whole catchment. The evaluation involves not only the soil properties, surface slope, etc, as in the evaluation for urban areas, but also the land use, such as agricultural lands or forested areas, natural protection areas, etc.. Different land uses require different evaluation regulations. Therefore, one such planning task already involves different domain knowledge. Similarly, this planning also involves massive evaluations of small areas distributed in a whole river basin.

1.3 Purpose of this study

Considering the complexity of the analysis of on-site stormwater infiltration measures in a single case and the large number of repetitions of this single analysis in a planning project, considering further the prospect of the application of on-site stormwater infiltration in Germany, even worldwide, and considering further the planning of the decentralised preventive flood control measures on a river basin scale with necessary empirical knowledge in different displines, it is strongly advised to develop a model or instrument to facilitate the planning with automation and in a flexible and systematic way. This is just the goal of this PhD study; that is, to develop a transparent open knowledge-based spatial decision support system for the analysis of decentralised rainwater management measures. The system is loaded with pre-defined knowledge for evaluating on-site stormwater infiltraion measures.

The pre-defined knowledge is derived from two practical projects carried in this study, that is, planning of on-site stormwater infiltration in the city of Chemnitz (Germany) and Emscher River Basin (Germany) respectively.

2. Methodology

2.1 Application of a knowledge-based system

The analysis for the optimal type of decentralised stormwater infiltration measure for a given situation in urban area or the soil water-retention potential in a agricultural land is a diagnostic process, i.e. evaluating a series of factors from different aspects against their critical values on different measures step by step. This process demonstrates the following characteristics:

1. The considered factors and their critical values can not be determined by pure theoretical considerations, but often depend on researchers' knowledge or experience and even on the current laws in relevant aspects, as well as public acceptance.

2. The availability of the data of all influencing factors can not be guaranteed and varies form project to project.

3. Some factors may represent different objectives that can not be unified. Therefore, a trade-off between objectives is often needed in an analysis. Such a trade-off is inherently subjective and relies on experience or a political viewpoint.

Because of these characteristics, the analysis for decentralised stormwater infiltration does not rely solely on fixed solution logic or algorithms, but relates to variable knowledge. This task can only be fulfilled by relevant experts or trained experienced engineers. However, due to the massive decisions for just one planning scenario, it is not a task that can be done manually. Likewise, it can not be automated with the help of the macro functions available in some general data processing programs such as MS Access or MS EXCEL due to its unstructural characteristics. In addition, because the knowledge involved in the decision procedure may probably change from one planning scenario to another or from one project to another, it can also not be encoded as a normal program to facilitate automation of the decision-making processes.

Fortunately, with the development of computer science and the artificial intelligence (AI) research, human knowledge or expertise in a specific domain can be programmed in a so-called expert system (ES) or knowledge-based system (see next chapter), which can then be used instead of the actual experts to provide expert-level advice or recommendations on the designed problem to users, such as planners or decision-makers. Moreover, an expert system is usually constructed so that its knowledge and the solution logic (i.e, the process of reasoning using the specified knowledge) are separated. The solution logic is then coded as a program while the knowledge is written in an easily understandable style as a supporting knowledge base (see next chapter). In this way, the specific knowledge can be easily modified to suit a change of situation of a problem domain for which the expert system is built, while the coded program of the solution logic remains unchanged. Hence, a knowledge-based system or an expert system is ideal for solving the specified decision problem of dynamic and unstructured properties.

2.2 Development of a GIS integratable "transparent" expert system tool

Although the expert system makes the change of the specific knowledge easier than in normal computer programs, the knowledge of an expert system is usually not open to users, let alone changed or modified by users. The change of the specific knowledge is usually done by so-called knowledge engineers. However, this does not meet the requirement of the problem defined in this study.

Therefore, to achieve the goal of this study, the first task was to develop an expert system tool with a simple but flexible knowledge formulation system. This tool should facilitate easy understanding and editing of knowledge in any of its specific knowledge bases so that decision-makers can directly manipulate the pre-defined knowledge of a built expert system. In addition, the tool should also be integrated into a GIS platform to be able to process the spatially oriented problem. Further for facilitating the preparation of the necessary spatial input data, a GIS extension program providing automation of some spatial operations, such as the intersection or union of many spatial distributions, is also necessary. The final integrated model should consist of "*ES tool + GIS extension + GIS platform*".

2.3 Development of a knowledge base for selection of on-site rainwater infiltration measures

After the development of the integrated model, i.e. ES tool + GIS extension (+ GIS platform), the empirical knowledge for selecting the on-site stormwater infiltration measures for two practical urban catchments, i.e. Chemnitz and the Emscher river basin, are analysed, and two case-specific decision trees or networks are developed. Based on these trees, two problem-specific knowledge bases were formulated according to the developed ES tool. The two developed decision support systems were then used in the practical planning projects and the results were explained in detail. In order to evaluate the sensitivity of the planning results on one of the criteria of the soil permeability Kf, the knowledge base was changed according to a series of designed Kf values; different planning results or scenarios then were obtained and evaluated.

3. Introduction to expert systems and their application

3.1 Expert systems and artifical intelligence

Expert systems are derived from a branch of computer science research called Artificial Intelligence (AI), which itself has been a formal area of research since the “Dartmouth Conference“ in 1956 by J. McCarthy, M. L. Minsky, N. Rochester, and C.E. Shannon. The general goal of AI is to build intelligent machines or intelligent computer programs that can understand languages, visually identify objects, solve problems with its own ‘thinking’, etc.. However, in striving towards this general goal, AI’s practioners still face some fundmental difficulties today, such as common sense reasoning and combinatorial explosion (Hopcroft and Ullman, 1979) of search spaces, although a great number of successes, such as advanced robots (capable of climbing stairs, dancing, etc.), sophiscated chess-playing programs (for example, IBM’s Deep Blue Supercomputer defeated world chess champion Garry Kasparov in May 1997) have been achieved since the start of AI research in the 1950s.

In contrast to AI’s general goal, expert systems usually concentrate more narrowly on certain realistic complexities (P. Jackson, 1999) that normally require a considerable amount of human expertise. In other words, an expert system will use available domain-specific specialised knowledge to solve corresponding specific domain problems. Through incorporating specialised domain-specific knowledge in its problem-solving strategies, expert systems usually avoid the above-mentioned AI difficulties. Furthermore, due to its focus on realistic complexities, expert system has great prospects for application in business, industry, engineering and research, and hence is becoming the most widely applied branch of AI. The process of constructing an expert system is normally considered to be ‘applied artificial intelligence’ (Feigenbaum, 1977).

The first expert system is normally considered to be the MYCIN, a rule-based system for knowledge representation and inference in the domain of medical diagnosis and therapy. However, the knowledge-based system (often used as a synonym for ‘expert system’) was released much earlier. The program Dendral (Edward Feigenbaum, Joshua Lederberg, Bruce Buchanan, Georgia Sutherland, 1967), which interprets mass spectra on organic chemical compounds is regarded as the first successful knowledge-based program for scientific reasoning. Since then, the “main-stream” path of development of AI has been the path of knowledge-based systems for achieving intelligent behavior in programs (Feigenbaum, at "AI's Greatest Trends and Controversies", 2000)

3.2 Building an expert system

Before further giving a detailed introduction to expert systems, the following text from Jackson (1999) gives a definition of an expert system:

> An expert system is a computer program that represents and reasons with knowledge of some special subject with a view to solving problems or giving advice.

3.2.1 Knowledge representation

Humans use usually abstract symbols or complex structures of symbols to describe objects and their relationships in the world. They reason things based on symbolic information. Therefore, lists of symbols or more complex structures of symbols are regarded as essential forms of representation of human knowledge in computer programs. However, to represent knowledge as symbol structures and use that knowledge to intelligently solve problems in a computer program, particular high level representation formalisms need to be designed. Such a representation formalism should be efficient to represent knowledge needed for a domain-specific expert system or systems. It should also support inference; that is, effective general solution logic can be encoded so that the computer (actually the encoded program) can use formulated domain-specific knowledge to reason or inference towards a soulution during each execution of a built ES.

The knowledge representation is in fact the key factor of the whole AI research. Most commonly-used knowledge representation approaches in AI are structured objects (semantic nets, frames), predicate logic and production rules (P. Jackson, 1999) (C. Alison, 1998). Of those, rule-based programming is a relatively simple but flexible method. Rules are in fact most commonly used by intelligent humans. "It is probably an axiom of artificial intelligence, and modern psychology, that intelligent behaviour is rule-governed" (Jackson, 1999).

Rules usually consists of a condition (IF part) and an action (THEN part), where the condition specifies a series of restrictions on the current status of a system, while the action specify a series of single actions to be applied if the condition is satisfied. For example,

> IF the temperatue is more than 35 degrees celsius THEN switch on the air conditioning,

IF the room is dim, THEN switch on the light,

These rules are especially effective for representing humans' heuristic domain-specific knowledge, where the solution is not based on strict logic inference, or in other words where the problems can not be sovled by a complete theory or axiom.

However, sometimes it is the properties and interrelationships of complex objects in the domain that determine the solution to a problem, while rules are not very convienient for representing knowledge about such objects and events and arbitrary relationships. Therefore, in today's AI practice, different formalisms are usually applied together to deal with the complexity of certain domain-problems, such as production rules along with structured objects and procedural knowledge.

3.2.2 Basic components of an expert system

In an expert system, human expertise or knowledge in solving a specific problem should be well represented in one or more special formalisms that computers can understand and apply appropriately in its reasoning process towards a solution to the designated problem. The well represented knowledge is saved as the so-called knowledge base, which is one of two basic components of an ES. The other basic component of an ES, called the inference engine, helps the computer understand the knowledge in the knowledge base. In other words, the inference engine interprets the knowledge and controls the computer reasoning during the excution of an ES. Aside from these two basic components, a so-called user interface is also important for eventually putting a constructed system into successful use, which provides a human-computer dialog for requesting data inputs and displaying the explanations of reasoning as well as results.

3.2.3 Building an expert system

As described above, the development of an expert system involves collecting and representing as much domain-specific knowledge as possible in its knowledge base. However, on one hand, human knowledge in a problem domain changes constantly. Over time, something which was once true might cease to be true; on other hand, more knowledge can be gathered with time from different sources other than the original one or because more information is now available that did not exist and hence has not been considered in the problem-solving stratigies at the beginning. Therefore, the knowledge base of a built ES constantly needs to be corrected, modified or updated. This introduces one important characteristic of the development of an ES, namely, exploratory programming (Alison, 1998). At first, a prototype of the system is programmed. With the testing and practical use of the prototype system, corrections or new considerations concerning the programmed knowledge may arise from both users and experts. The system is then modified. Of course, it will be modified again and again afterwards.

When following conventional programming, the modification of the knowledge of an ES means carefully finding all locations in whole source codes where the change of the knowledge consequently causes the code to change. It may only require some modest changes, such as changing a constant K=3 to K=3.8, or changing a numeric function x=b*cos(a) to x=b*sin(a/c), but it may also change the fundamental structure of the code. This is actually very tedious work and very costly when the knowledge base is relatively large. Needless to say, the changes will frequenly occur. Fortunately, the AI researchers have found another way to develop expert systems, namely, separating the knowledge base and its solution logic. The solution logic is then encoded and compiled as an excutable program, while the knowledge base is written in certain special fomalisms and simply loaded or read into memory and interpreted by the 'fixed' solution logic during an excution of the ES. In this way, the changes can be made to the knowledge base repeatedly, which can be actually facilitated by a so-called knowledge acquisition system. Also, if it is ncessary to change the solution logic for better performance, the knowledge base or bases (if many domain-specific systems have been developed) could be kept intact.

The encoded solution logic is actually the above-mentioned basic component of an ES, i.e. inference engine. An inference engine along with an empty knowledge base is usually called an ES shell. Because one can use this shell to develop an ES simply by transforming the necessary domain-specific knowledge into an empty knowledge base, it is also called an ES tool.

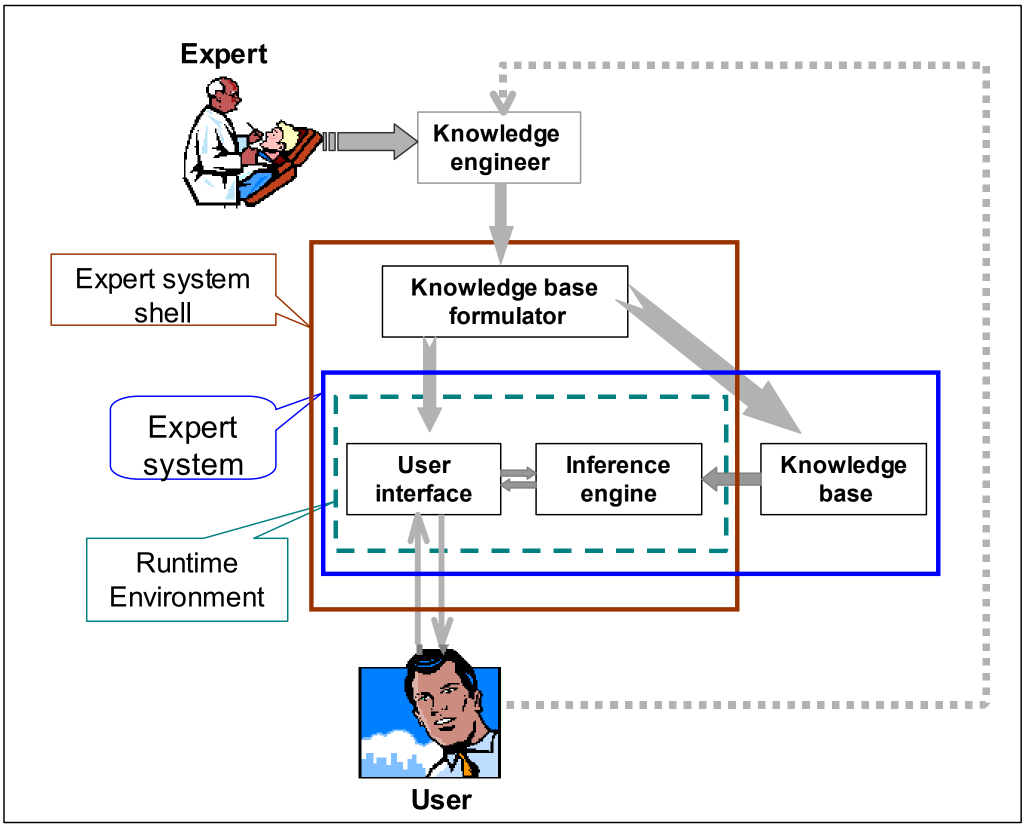


Fig.5 Schematic presentation of the relationship of components of an expert system and its shell

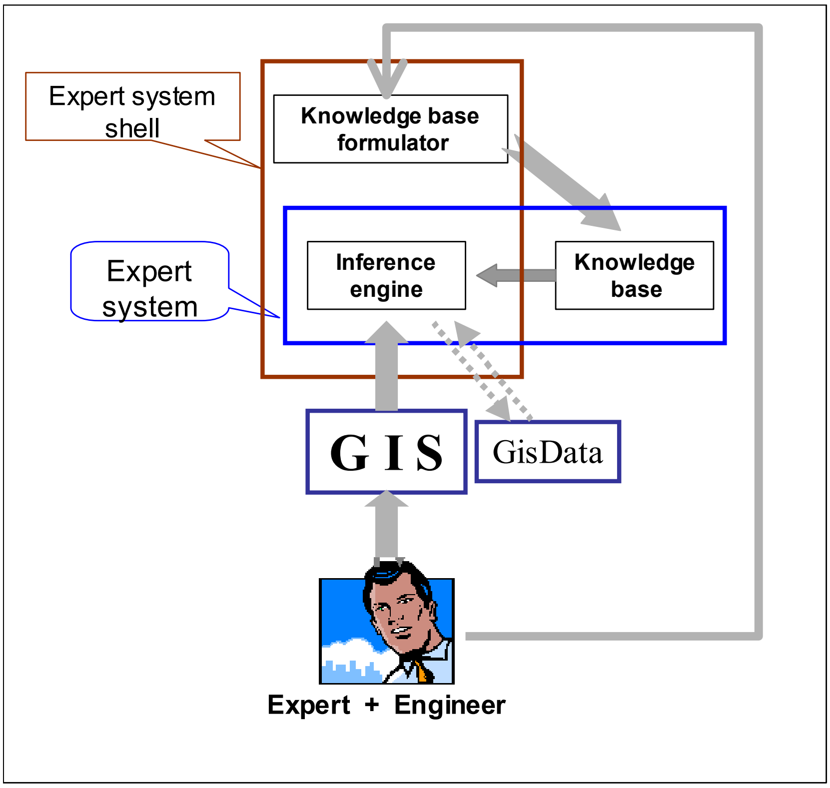


Fig.5a Schematic presentation of the relationship of components of the expert system and its shell developed in this study

Figure 5 demonstrates schematically the relationship among domain experts, an ES and users. It also shows an ES shell and its basic components.

Figure 5a demonstrates schematically the relationship among domain experts or engineers, an ES and its shell developed for the specific purpose in this study, that is, an expert system and its shell are integrated into a GIS platform and used for automatic answers of a large set of inquiries other than in conventional human-computer dialog style. In this case, the dialog interface is not present and the inference engine accesses the data files directly for input and output. The engineers directly develop their knowledge bases without help of the knowledge engineers.

Because an inference engine is independent from concrete knowledge, an ES shell can be used to develop different domain-specific expert systems only if the different domain-specific knowledge is represented in the formalism the inference engine understands. It should be stressed that the inference engine is not independent from the knowledge representation formalism because the solution logic of an inference engine is just encoded for its specially designed formalism.

In fact, aside from high-level expert system shells, expert systems can also be programmed by using so-called AI programming languages, such as LISP (invented by John McCarthy in 1958) and Prolog (developed by Alain Colmerauer in 1972), which are characterised as supporting list computation and exploratory programming. By programming from scratch, the developer must design his own knowledge representation formalisms and write the inference module to interprete the stored domain-specific knowledge represented in the designed formalisms. In this way, the developer must make more programming efforts but enjoys more freedom in representing his domain knowledge. In fact, many of today's object-oriented programming languages are also suitable for programming expert systems or shells because the classes and their instances can be used to efficiently represent knowledge. For example, one very popular expert system shell CLIPS (see next chapter) was developed in C language, which support production rules and structured objects for representing knowledge. The developed expert system tool in this study was also programmed in Visual C++ and Visual Basic.

3.3 Literature review on expert system application in water science

Expert systems, the most applied AI branch, can be found nearly everywhere, from financial management to flight schedule planning, from diagnosing diasease to oil exploration, from student learning systems to space shuttle maintainace. However, the typical application domains for expert systems are mainly:

- Diagnosis/classification
- Mornitoring and control
- Education
- Planning
- Design

In this section, reviews are not given to the overall use of expert systems, but rather limited to those from water science, especially from urban stormwater management, among which efforts to integrate an ES with GIS in on-site urban stormwater management will be reviewed in detail. For the clear definition of the literature search area, the search in the ASCE (American Socity of Civil Enginers) database and in TIB/UB Hannover (library of the University of Hannover, Germany) for research or applications after 1990 (except Nr.1) and on the use of expert systems or knowledge-based systems was carried out. Among the found items, the items in hydrology and water resources area were listed in following two classes (Tab. 1, 2).

Table 1 Applications with only an expert system involved

Nr.	Author	Year	Title
1*	L. Fuchs, H. Neumann	1988	Ein lernendes Expertensystem zur Steuerung städtischer Kanalnetze
2	Lam, D.C, et al	1990	A Knowledge-Based Approach To Regional Acidification Modelling
3	Teutsch G., Barczewski, B, et al	1991	Bewertung von Grundwaserprobenahmetechniken zur Erkundung und Überwachung von Altlasten
4	James Crum and Michael Mulvihill	1991	An Expert System for the Planning and Design of Flood Control Channels
5	R.L. Doneker, G.H. Jirka, P.J. Akar,	1991	Expert Systems for Mixing-Zone Analysis and Design of Pollutant Discharges
6	S. Y. Liong, et al	1991	Knowledge-Based System for SWMM Runoff Component Calibration
7	Nir, D.		Expert systems for agricultural water systems
8	William L. Magette *et al*	1992	Expert System for Agricultural and Water Quality Management

9	Albertsen, Manfred	1993	ein Expertensystem für Fluid-Rock-Interaction-Probleme in Wasser- und Erdölbohrungen
10	Mark Svendsen and Chandra Balachandran	1993	A Flexible Expert System Based Typology of Irrigation System Types
11*	P. Stephen Lundgren and Michael E. Barber	1993	Development of an Expert System for Urban Runoff
12	Katherine Hon	1993	Including Expert System Decisions in a Numerical Model of a Multi-Lake System Using STELLA
13	Tiao J. Chang and David Moore	1994	Reservoir Operation by the Use of an Expert System
14	Yoon, J. Padmanabhan, G.	1995	Heuristic Knowledge-Based Tool for Rainfall Synthesis, Runoff Estimation and Hydrograph Generation
15	James M. Crum	1996	A Knowledge Based System for the Design of Open Channels
16	Anne Shepherd, et al	1996	Water-Supply System Operations: Critiquing Expert-System Approach
17	T. J. Chang, et al	1996	Development of an Expert System for Daily Drought Monitoring
18	Xin X. Zhu	1996	Expert System for Water Treatment Plant Operation
19	H. K. Lee, *et al*	1997	Fuzzy Expert System to Determine Stream Water Quality Classification from Ecological Information
20	Chang, J. Moore, D.	1997	An Expert System Approach for Water Management in Case of Drought
21	Carlos León et al	2000	EXPLORE–Hybrid Expert System for Water Networks Management
22	Rabee M. Reffat	2001	Expert System for Environmental Quality Evaluation
23	Chau, W. Chen, W	2001	An Example of Expert System on Numerical Modelling System in Coastal Processes
24*	M. Gómez, H. Sánchez, and S. Vázquez	2002	A Simple Expert System for Initial Size Estimation of Detention Basin

25	Margaret A. Hahn, Richard N. Palmer, P.E.	2002	Expert System for Prioritizing the Inspection of Sewers: Knowledge Base Formulation and Evaluation
26	Rolf Gimbel ; Hans-Dieter Kochs ; Jörg Petersen	2002	Modellierung mehrstufiger Trinkwasseraufbereitungsanlagen mittels eines expertensystem-basierten Simulationsmodells (METREX) am Beispiel von Oberflächenwasser
27	Ramesh S. V. Teegavarapu, et al	2003	MIST (Model Identification and Selection Tool): A Knowledge-Based System for Selection of Water Quality Models
28	Yang, S., *et al*	2003	A Model of Intelligent Interpreting Soil Erosion Based on Geographical Knowledge
29	Sojda, Richard S	2003	Decision Trees within Expert Systems for Modelling Ecological Knowledge Related to Water Level Management for Palustrine Wetlands
30	Abu Noman M. Ahsanuzzaman	2004	Simple Expert System for Evaluation of Nutrient Pollution Potential in Groundwater from Manure Application

* in field Nr. means the application is relevant to urban hydrology

Table.2 Applications with integration an expert system with GIS

Nr.	Author/developer	Year	Application/Research Title
1*	Simonovic	1993	Flood Control Management by Integrating GIS with Expert Systems, in Application of Geographic Information Systems in Hydrology and Water Resources Management
2*	Daene C. McKinney, *et al*	1993	Expert Geographic Information System for Texas Water Planning
3	Lam, D.C.L. and C. Pupp	1996	integrating GIS, Expert System, and modelling for State-of-Environment Reporting
4	ZHU, X.; HEALEY, R.G.; ASPINALL, R.J	1998	A Knowledge-Based System Approach to Design of Spatial Decision Support Systems for Environmental Management
5	Sven Zimmermann	1999	Regionalisierung in der Hydrologie
6	Comas, J *et al*	2002	The STREAMES Project: Linking Heuristic and Empirical Knowledge into an Expert System (ES) to Assess Stream Managers
7	MERTA M., SEIDLER C., et al	2003	Das wissensbasierte System FLAB als Instrument zur prozessbezogenen Raumgliederung in mesoskaligen Gebieten

8	Leibundgut, ch. Eisele, M.	2004	Catchment Oriented Assessment of Runoff and Solute Dynamics - Development of the Assessment Procedure 'Hydrological Quality' into an Expert System for River Basin Management

* in field Nr. means the application is relevant to water management or planning

From the literature review, it is clear that the expert systems were widely used in water science, too. Most of these applications concentrate on a specific problem area, and their knowledge is not changeable by users. In addition, although there are several cases involving an integration of an expert system with GIS, nothing is found in application for the planning of on-site stormwater management, let alone the integration of a transparent and open expert system in this direction. This is further confirmed by the following table, which gives a review of the recent overall research on stormwater management planning.

Table 3. Researches or practices in urban stormwater management planning

Nr.	Author	Year	Title
1	Steffen P. Meyer Tarek H. Salem, John W. Labadie	1993	Geographic Information Systems in Urban Stormwater Management
2	Thomas R., Sear Ronald L. Wycoff	1993	Selection of Optimal Best Management Practices (BMP's)
3	Daniels, K.	1994	Integral or Integrated Planning - Only One for Future?
4	Glenn A. Bottomley	1995	Using SWMM in Urban Stormwater Master Planning
5	William N. Lane,	1995	Blue-Green Revisited: Integrating Stormwater Management Into the Urban Planning and Development Process
6	David C. Froehlich	1996	Searching for Optimal Combinations of Stormwater Detention Basins
7	James A. Bachhuber	1996	A Decision Making Approach for Stormwater Management Measures—A Case Example in the City of Waukesha, Wisconsin
8	Larry A. Roesner, P.E., Richard M. Howard, P.E., *et al*	1997	Integrating Stormwater Management into Urban Planning in Orlando, Florida
9	Ben Urbonas, P.E.	1997	Design & Selection Guidance for Structural BMPs

10	K. K. Chin, K. Y. Ng, S. W. Lee, M. Hasni, A. Choo	1998	Urban Renewal and Stormwater Management Planning in an Urban Center
11	Fabian Papa, Barry J. Adams, Yiping Guo	1998	A Family of Analytical Probabilistic Models for Urban Stormwater Management Planning
12	Bolder, J., Lindiger, T., Zech, H. *et al*.	1998	Planung und Bau des Retentions-Bodenfilters 'Rieselfeld' in Freiburg
13	Sieker H., Bandermann S., Zimmerman U.	1999	Geographical information systems used as a tool for decentral stormwater management
14	SIEKER, H.	2001	Generelle Planung der Regenwasserbewirtschaftung in Siedlungsgebieten

4. Development of a flexible expert system tool

4.1 Summary

Since the appearance of EMYCIN (Bill VanMelle, 1979), a program that demonstrates the generality of MYCIN's representation of knowledge and style of reasoning, numerous expert system shells have been developed. However, an existing shell was not directly used to build the expert system for the evaluation of on-site stormwater infiltration in this study for the following reasons;

1) An existing shell usually has a set of its own fixed interfaces, which make the shell difficult to integrate with other parts of the overall model to be constructed, for example, the GIS extension program and GIS platform in this study. In fact, most existing shells do not provide the possibility of direct integration with the GIS environment. The integration can only be realised through embedding an existing independent shell in own programs.
2) Most existing shells (for example, the expert system shell CLIPS, see **4.2**) do not meet the extent of simplicity required by the specified tasks in this study (see chapter one and Fig. 5a), because this study does not aim to build an expert system that is regarded as an authoritative consulting system, but to build an open, transparent, knowledge-based decision support system, whose knowledge could be modified by engineers or decision-makers themselves according their own knowledge or demand,
3) most existing shells are for commercial use and expensive, hence not suitable for this research purpose.

Detailed enumerating commercial shells are referred to in books by Ram D. Sriram (1997) or by Peter Jacson (1999), etc..

Nevertheness, the development of the shell “Flext” did originate from an existing expert system shell CLIPS (C Language Integration Production System), because it supports flexible knowledge representation and is able to be modified and embedded in other applications’ code and is free for use in research activities.

At first, CLIPS was adapted and embedded in a VC++ program to provide a runtime environment with a reasoning module. This runtime environment, along with a new graphical knowledge base formulator (programmed in Visual Basic 6.0), forms the expert system tool FLEXT.

The knowledge base formulator enables the formulation of rules (conditions and actions) in natural language and in a structural system, i.e. as trees or networks of nodes. It also enables the design of question-specific graphical interfaces. In addition, the structure of a rules system can be displayed and edited in a graphical interface.

The runtime environment includes the following procedures or functions:

1) translating the Flext-style rules system into CLIPS format rules;
2) adapting the CLIPS inference to the structured rules system (structured as trees or networks of nodes) and making it capable of processing question-specific graphical interfaces;
3) additional functions such as input/output from text or database files, database searches, execution of external programmes, etc..
4) repeated execution of an expert system with directly reading inputs from a database, such as a database that stores the attribute of a GIS feature. This is actually the main motivation to develop this expert system tool: to develop GIS-oriented expert systems.

However, through testing the application in some practical projects, it was found that the adapted runtime environment, although its adapted reasoning module proves successful in reasoning the tree-structured or network-structured rules system, still needed to be enhanced so that it would become more direct (regarding its concentration to the specific problem domain in this study), more integrated (regarding its integration with the knowledge base formulator mentioned above) and more flexible (regarding its more flexible output function both during and at the end of the execution of an ES), more powerful (regarding its possibility of visually tracing the reasoning process step by step). Therefore, a new reasoning module was developed in Visual Basic 6.0 and encoded along with the knowledge base formulator as a single program. This self-developed reasoning module is completely different from CLIPS in data structure and programming technology, although the inference philosophy, i.e., the rule cycling principles in an inference process, is similar to the one in CLIPS or any other forward-chaining rule-based system.

In the following sections of this chapter, CLIPS and the developed tool Flext (with both the inference engine adapted from CLIPS and the self-developed one) are described in turn.

4.2 Introduction to expert system shell CLIPS

CLIPS (C Languaged Integration Production System) is one of most popular expert system shells. Its prototype version was developed in the Spring of 1985 by the AI division of NASA and has been updated continuously ever since. This transforms CLIPS from being a tool only for training purposes to a useful and widespread tool for the development of expert systems. Today it is public domain software.

4.2.1 Representation of knowledge

CLIPS uses facts, objects and global variables as a basic format for representing information. A fact is a list of data of finite size consisting of any so-called primitive data type-integer, float, symbol, string, etc., except that the first piece of data in the list should be a symbol. The list is enclosed by a set of parentheses. For example:

(groundwater depth is 2.0 m)
(stormwater-measures “MRS” “MRE” “MV”)

Data in such facts is accessed by position when processed; these facts are therefore called ordered facts. There are also non-ordered facts, i.e. so-called deftemplate facts, in which each data field or slot has a name and is also enclosed by a set of parentheses. The first field of a deftemplate fact is the name of the deftemplate. For example:

(field-data (kf 168 m/h) (groundwater-depth 2.0 m) (surface-slope 0.1))
(field-data (groundwater-depth 2.0 m) (kf 168 m/h) (surface-slope 0.1))

Hence the data in deftemplate facts is accessed by their field names, so the above two facts are identical regarding the information they represent.

An object is an instance of any primitive data type as well as so-called user-defined classes. For example:

Class	**Object (**printed representation**)**
Integer	18
String	“OK”
MRE Measures(user-defined)	[MRE1]

Primitive type objects are referenced simply by giving their value, while the instance of a user-defined class is referenced by its address or name. An instance name is formed by enclosing a symbol with left and right brackets. As in above examples, [MRE1] is the name of an object MRE1 which is an instance of user-defined class MRE Measures.

An instance of a user-defined class usually inherits properties specified in terms of slots (named single field or multi-fields values) and behaviours specified in terms of procedural code from its class. For example:

The class MRE Measures may have a slot “life-span” and its value for this slot might be “10 year”. The class MRE Measures may also have a piece of procedural code for calculating the construction costs of its instances. The object MRE1 has therefore the behaviour, also called message-handler, which calculates its own construction cost. If the message “calculate cost” is sent to the object MRE1 in designed steps of a reasonning process during execution of an expert system with this object, it will respond with a result, i.e. the cost of its construction. Of course, the object MRE1 has the property of 10 year life-span too, which is directly inherited from its class.

A global variable is defined the same way as in other conventional programming languages; that is, a global variable can be accessed anywhere in the CLIPS environment at runtime.

In addition to representing information or factual knowledge, CLIPS also provides heuristic and procedural paradigms for representing knowledge. The production system (If---Then rule) is used for representing heuristic knowledge, i.e. heuristics, “rule of thumb”, which specifies a set of actions to be performed for a given situation in which all conditions specified in the rule are satisfied. Conditions can be specified in a so-called pattern consisting of a set of restrictions. Patterns can then be used during the system execution to determine whether conditions of some rules have been satisfied by the current facts or objects. This process of matching facts or objects to patterns of rules is called pattern-matching. In runtime, the inference engine of CILPS will continuously run pattern-matching against the current state of the facts and objects.

Originally, CLIPS uses primarily a forward chaining rule language representing knowledge. With continuous updates, the procedural paradigm, as well as object oriented programming methodology, are introduced. As already mentioned before, the procedural code attached to a user-defined class allows users to define behaviour of the objects of that class. CLIPS also allows users to define functions which are procedural pieces of codes too.

Based on this knowledge representation methodology, CLIPS provides an set of commands for users to define facts, classes, rules, functions, etc. in building their knowledge base. Following example shows how a rule is defined in CLIPS syntax.

```
(defrule normal-engine-state-conclusions ""
(declare (salience 10))
(working-state engine normal)
=>
(assert (repair "No repair needed."))
(assert (spark-state engine normal))
(assert (charge-state battery charged))
(assert (rotation-state engine rotates)))
```

This rule specifies a very simple pattern (a sets of restrictions), which checks whether the ordered fact (working-state engine normal) exists in the current fact-list and if so, this rule will be applicable and eventually executed (it depends further on whether there are other rules applicable and the priority of this rule and other rules). The action of this rule is asserting four new ordered fact.

4.2.2 Basic cycle of rule execution

In CLIPS, a knowledge base can be partitioned into modules if necessary. In runtime, all modules are put in a so-called focus stack. However, there must always be only one module in current focus. At start of an ES, the module in current focus is always the module MAIN. If this module is not explicitly defined in the knowledge base, it will be automatically created by the system.

a) The rules in current focus module will be pattern-matched for activation and the activated rules are put into a so-called agenda, which is similar to a stack. The placement of activated rules in the agenda depends on the salience, which can be specified in the rule definition syntax. The higher the salience of an activated rule, the higher it is in the stack. For rules with the same salience, their placement in the agenda will be decided according to the conflict resolution strategy. There are in total seven strategies. The most common strategies are the breadth strategy and the depth strategy. With the breadth strategy, the newly activated rule will be put below all rules of the same salience on the agenda, while with the depth strategy, the newly activated rule will be put above all rules of the same salience on the agenda.

b) The rule on the top of the agenda is selected for execution. Directly after the execution, step a), the pattern-matching process will repeat. Therefore, as a result of the execution of a selected rule, some rules may become activated and some activated rules in the agenda may be deactivated and removed from the agenda. If there are no more rules in the agenda, the module with current focus will be removed from the focus stack and the next module in the focus stack will get the current focus. The current focus can be also changed as a result of the execution of an activated rule when the **Return** function is used as an action of that rule.

The pattern-matching a) and the execution of selected rule b) will repeat until there are no more modules in current focus or the so-called rule firing limit is reached.

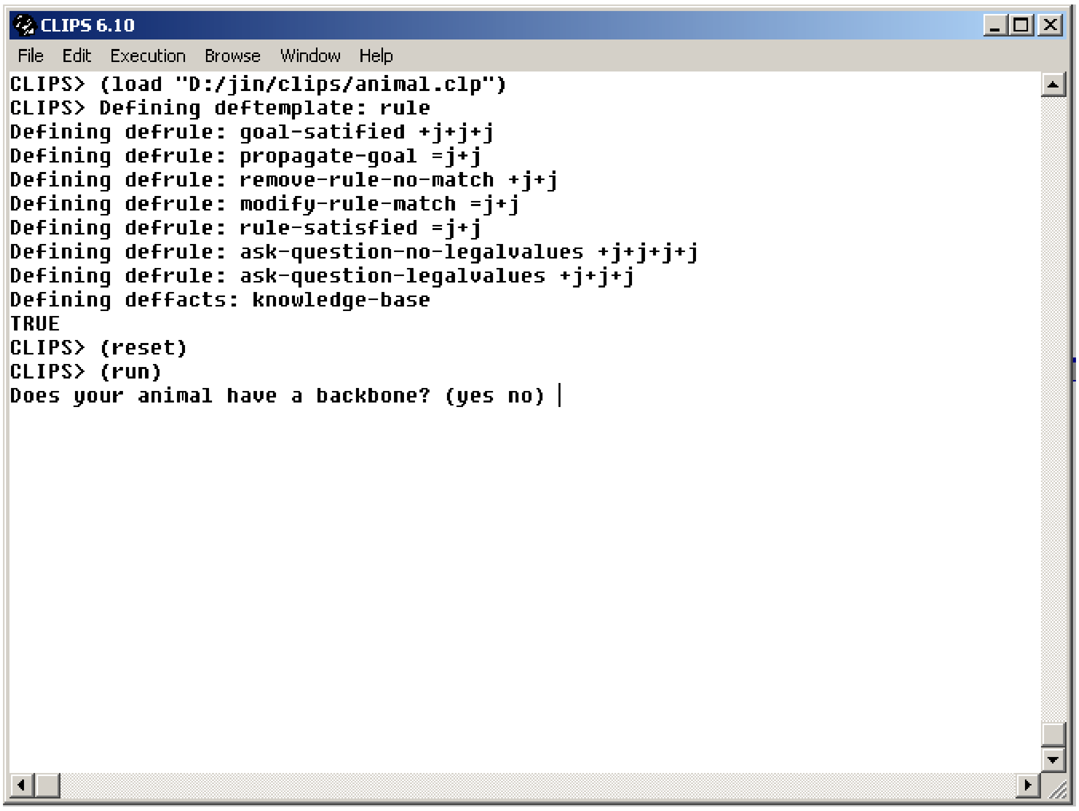


Fig. 6 The interface of an exemplar execution, CLIPS windows version

4.2.3 Formulation of knowledge base and running an expert system

CLIPS-version 6.10 provides a platform to run a specific expert system in Windows operation system. A knowledge base can be first loaded and then be browsed according to its construct type. One can set watch windows for facts, global variables, instances, etc. Figure 6 shows the CLIPS interface of executing an expert system.

This programme also provides an editor for writing a knowledge base. However, it appears not much different than a pure text editor. Knowledge (rules, facts, functions, etc.) should be written in relevant syntax format as pure text.

For more detailed information on CLIPS, see documentation on CLIPS.

4.2.4 Remarks on CLIPS

CLIPS is a powerful tool for developing expert systems as it applies both production rules and structured objects as knowledge representation formalisms. In addition, CLIPS is also capable of processing numerical computations, which is usually not easy in AI programming languages. However, its knowledge formulation interface is relatively simple and its knowledge representation system and syntax are rather complicated.

It is certainly difficult and time-consuming for people (experts, engineers) without training in CLIPS to formulate knowledge bases in their research. Even for a so-called knowledge engineer who has special training in CLIPS will still find it rather confusing to manage how rules react with each other after he has built a knowledge base of hundreds of rules. Which rules will be fired next if certain rules have already been fired ? This confusing rules system could lead to unexpected results in application.

The relatively simple dialog interface as shown in figure 6 is surely a negative factor for the success or even acceptance of a developed expert system by its final users. To solve the second problem, one can integrate CLIPS into one’s main programme and try programming a user friendly interface through conventional programming language, yet the first problem for acquainting with CLIPS syntax and manipulating a large amount of rules (if the system is such) remains. In summary, the main disadvantages of CLIPS considering that the experts or engineers use CLIPS directly to build a problem-specific knowledge base for the task in this study are:

1) unconventional syntax
2) unclear logic relationship between rules in a knowledge base
3) dialog interface too simple
4) low possibility for integration with other programmes

4.3 Development of an expert system tool Flext

4.3.1 Knowledge representation

In Flext, the knowledge representation is greatly simplified and formulated in a natural language. As opposed to facts, objects and global variables used in CLIPS, Flext uses only global variables to represent basic information, rules and functions to represent complex knowledge. The condition of a rule is simply written in conventional logic mathematical expressions. Functions are also written in a conventional way. Actions of a rule are formulated by using a few keywords, like **Set**, **GoTo**, **Conclusion**, **Run**, **OpenDatabase**, etc. together with conventional mathematic operations and functions. For example:

> **Condition:** *((x>0) && (y<=100)) || ((a+b)=18)*
> **Actions :** ***Set*** *z=((x*y)+a),c=(y-b),d=9,test= "OK !"*
> ***Print*** *"Now run an external model"*
> ***Run*** *d:\test.exe*
> ***GoTo*** *nextnode*

Where *x, y, z, a, b, c, d* are declared as Number and *test* as String, *test.exe* is an external programme, *nextnode* is a name of another node (see below) in the knowledge base. **Print** sentence just relays a message to the user.

In fact, in the action-part of a rule, a set of single actions such as complex mathematic computations, database operations (limited), etc. can be sequentially executed. This means procedural knowledges can be also used in Flext. However, they are rather simple because the loops syntax and if–then-else syntax as well as complex data structure such as array are not supported in formulating actions of Flext rules.

For more complicated or comprehensive knowledge such as professional simulation models, Flext provides the opportunity to integrate any external executive programme by using keyword **Run** in an action of a rule as above example demonstrates. The necessary data exchange between the external programme and the expert system during an execution can occur with files. For example, if the cost of a rainwater management measure is needed in the decision process, an external program calculating the cost of different measures may be written in Visual Basic, and then referenced in the relevant actions of rules. The procedure is as follows:

- first specifying the output of relevant pieces of information which are necessary input parameters for the VB programme (for cost calculation).

- Specifying the execution of the VB programme
- Specifying the input from the output results of the VB programme.

Because the expert system is suspended and waits for the end of the execution of an external programme (the cost calculation programme in this case), the introduction of an external procedural programme does not introduce further complications on the inference process as specified in the knowledge base.

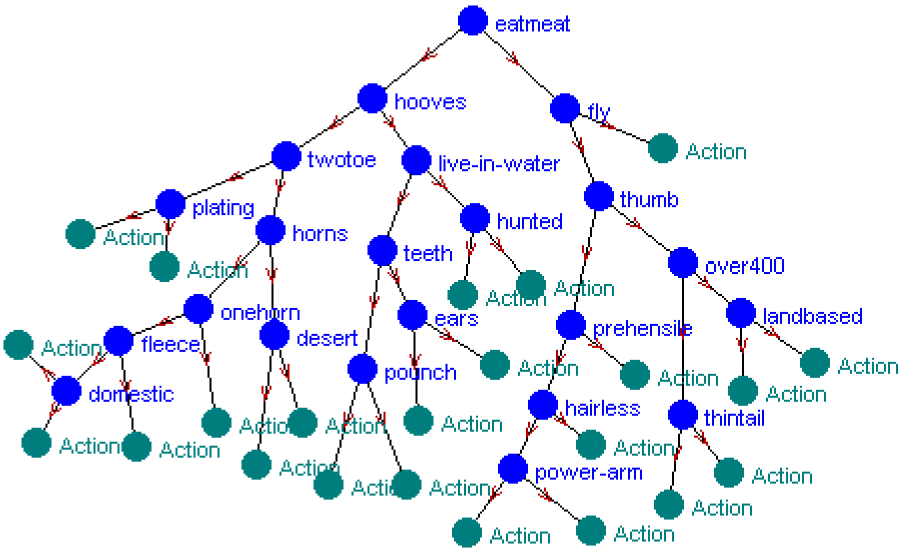


Fig. 7 Rules system as a tree of nodes

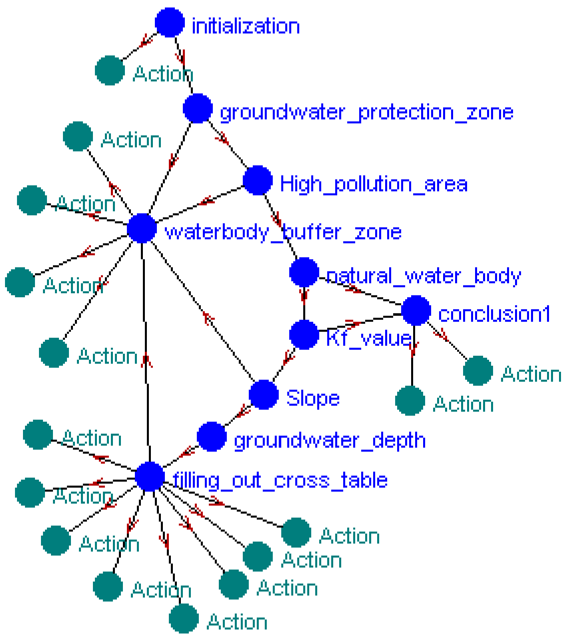


Fig. 8 Rules system as a network of nodes or multi-trees

4.3.2 Structure concept

The most important characteristic of Flext is the grouping of relevant rules into nodes. The whole rules system can be grouped into nodes and the logic relationship or the inference logic between nodes can be either explicitly expressed through the keyword **GoTo** in the action parts of rules of a node or implicitly expressed through the condition parts of rules of a node. It is particularly meaningful that the explicit logic can even be visually displayed in the graphical interface of the knowledge formulator of Flext. So, an rules system in Flext is constructed as a tree, trees or networks of nodes (Fig. 7, 8, att. 9).

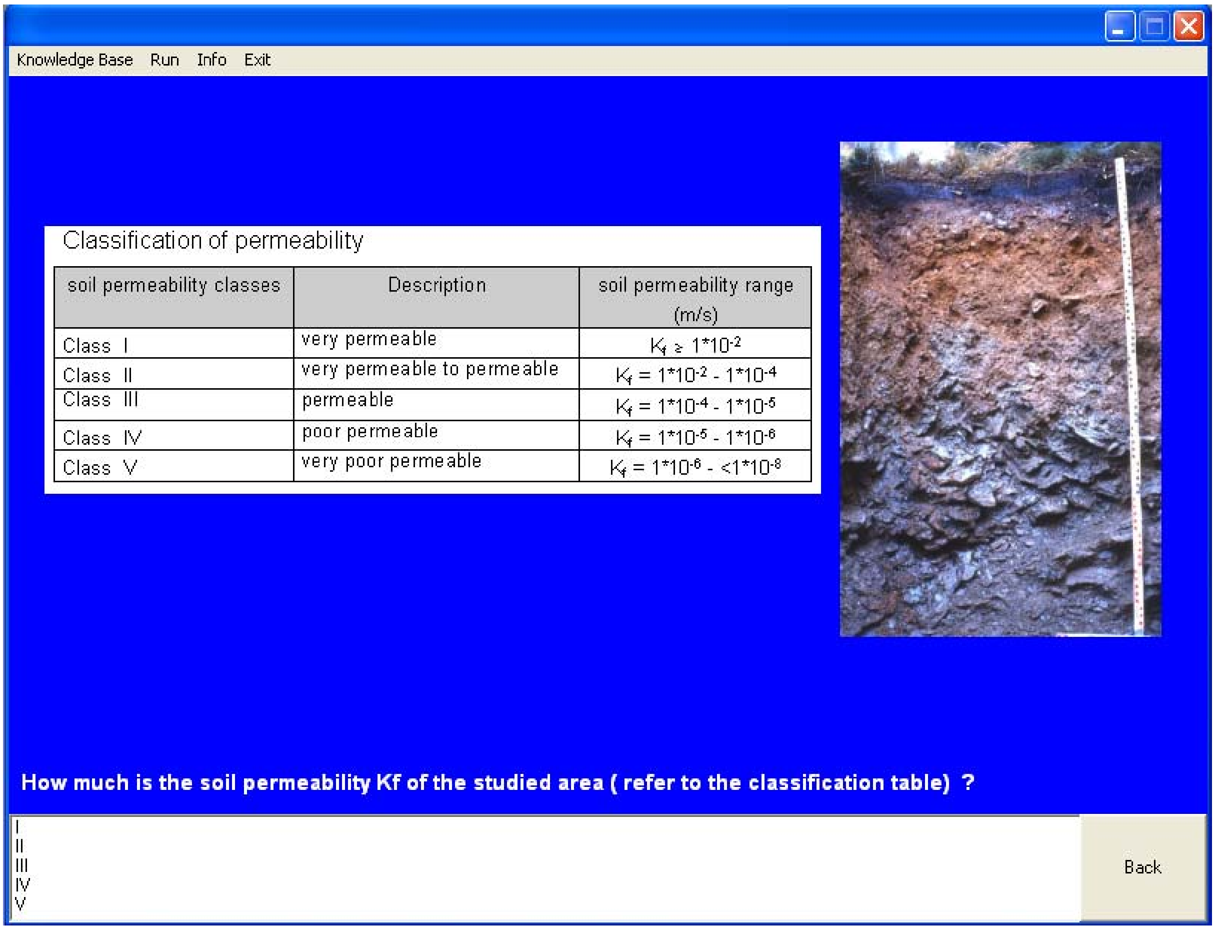


Fig. 9 An example of a dialog interface for asking a question

Another characteristic of Flext is the systematic process of questions. Questions with graphical interfaces can be defined in nodes. The knowledge base developer needs only to define the necessary questions in relevant nodes and their desired order (if there is more than one question in a node). When the question is asked and how many questions are actually asked in an execution is processed by the Flext inference engine. It also facilitates displaying the question-specific graphical interface (Fig. 9). The question-specific graphical interface can be directly desigend in knowledge base formulator. Moreever, any standard rich text documents can also be referenced as question-specific interfaces (att. 7).

4.3.3 Interpretation possibility

One of the important functions that an ES must possess is interpretation, which explains the user the reasoning behind a decision. Flext runtime interface provides an opportunity for output hints, remarks and explanations. The keyword **Print** can be used in actions of a rule to halt a running process and output the desired information. At the command of the user, the running process will continue. In one **Print** sentence, as many constants (text or number) or values of variables as necessary can be printed (att. 8). For example, one can use variables to store all questions currently being asked and their corresponding answers and use a **Print** sentence in the action parts of rules to print the values of those variables in desired steps of a reasoning process.

Print “This is a sample remark”, hints

This syntax indicates the inference engine to halt and print a piece of text and the content of the variable *hint*.

Similar to the question-specific interface, a standard rich text document can be displayed as message during reasonning process.

4.3.4 Inference mechanism of nodes, rules, questions

4.3.4.1 Adaptation of the CLIPS inference engine

In this adaptive version of the inference engine, the basic rule cycling is the same as described in section 4.2. However, since the rules system in the Flext knowledge base is organised as trees of nodes or networks of nodes and the question systems have been introduced in Flext, extra procedures are added to adapt the CLIPS inference mechanism so that it is capable of processing these new functions.

The Flext knowledge base is not partitioned into different modules and therefore, all rules will be in current focus module “Main” immediately after the start of an ES. However, in pattern-matching for the activation of rules, only rules from active nodes can possibly be activated. All rules from inactive nodes are identified so that their conditions can not be satisfied as long as their nodes remain inactive. At the beginning of a system execution, all root nodes of a Flext knowledge base are identified as active nodes, while others are inactive nodes.

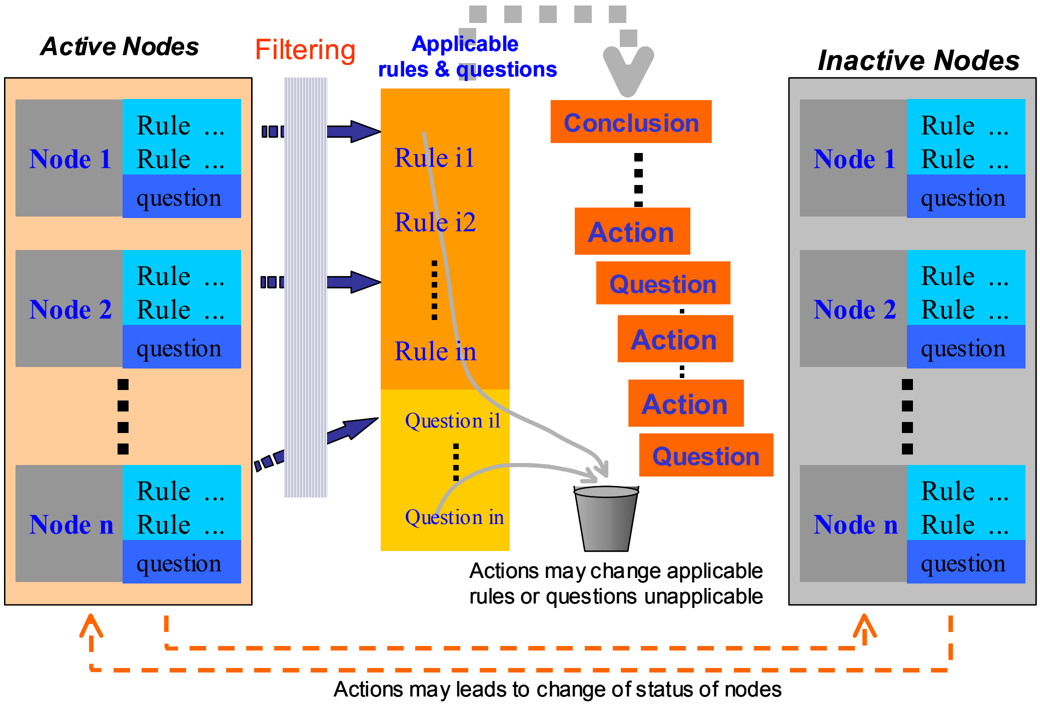


Fig. 10 Firing mechanism of rules and questions in Flext (adapted engine)

As in CLIPS, during the system execution, the conditions of rules are checked (like pattern-matching in CLIPS) for the activation of rules, and activated rules are put into the agenda. However, the placement of activated rules in the agenda depends simply on their sequence in the edit table of the knowledge base formulator. After the checking procedure, the rule at the top of the agenda will be selected for execution. As a result of the execution, the currently active node (to which the executed rule belongs) may become inactive while an inactive node may become active due to the **GoTo** function being used as an action of the executed rule. The checking procedure will be repeated after execution of the selected rule and again the rule from the top of the agenda will be selected for execution. This checking-execution process will be repeated until there are no more rules in the agenda. Questions in active nodes will then be asked in turn. However, after each question's input, the checking and execution process will be repeated before the next question is asked.

The checking-execution-query process will be repeated until a conclusion is reached. In Flext, a conclusion can be specified in actions of relevant rules. The system will be also stopped when there are no more activated rules in the agenda and there are no more questions in currently active nodes. In this case, the execution of the system is ended unsuccessfully, i.e. without results.

4.3.4.2 Development of an independent inference engine

As mentioned In the summary (see **4.1**), the adapted inference engine needs to be enhanced because it has the following shortcomings:

1) it is not convenient to manipulate the internal reasoning process in such way that any system information can be outputted at any point during the reasoning process.
2) communication between the two modules (knowledge base formulator and runtime environment) is not convenient
3) the CLIPS functions library is a free resource and hence the quality can not be completely assured.
4) Integration of the adapted inference engine with other programmes is limited.
5) The possible extension of the inference functionality in the future is limited.
6) The CLIPS inference engine is for a very wide range of knowledge representation formats, hence it is not so economical (regarding the comsuming working memory) compared with an independent specific for the tree-structured or network-structured rules system with the information are represented simply by variables.

Considering these disadvantages on one hand and the understanding of the general rule cycling concept from CLIPS on the other hand, an independent, very concentrated inference engine has been developed in Visual Basic 6.0, combined with the knowledge base formulator as one executive programme.

Rules cycling concept

The basic philosophy of this independent inference engine is still similar to the one in CLIPS or any other forward-chaining rule-based systems; that is, the system is driven by data. After startup of the system, the inference engine continuously monitors the change in values of variables in the knowledge base due to input or an action of a rule (at start of the system, all variables are null). If there are changes, the inference engine then checks related rules in the active rule list and checks all stand-by functions to determine whether the conditions of the active rules are satisfied or whether there are changes in the arguments of the stand-by functions. A stand-by function is actually an equation, the left side of which is a variable and the right side of which is a mathematic function. It does not reside in any nodes. The mathematic function will be computed as long as there is change in its arguments. Active rules are rules from active nodes. At system startup, all root nodes of a knowledge base (i.e. a decision tree or a network of nodes) are active nodes. The status of nodes will dynamically change during the inference process due to the execution of the rules. In order to run the check process systematically, a specific variable storage structure, rule condition structure and function structure are constructed (see next section).

The rules whose conditions are satisfied are called applicable rules and are put into a stack. The sequence of the applicable rules in the stack depends on the sequence of the rules in the edit table of the knowledge base formulator. The stand-by functions are called applicable functions if there are changes in their arguments and are put into an another stack. The inference engine executes all applicable functions first and checks the active rules and stand-by functions that relate to the left-side variables of the executed applicable functions again for applicable rules and applicable functions. This process will be repeated until there are no more applicable functions. Then, the applicable rules at the top of the applicable rules stack (similar to the agenda in CLIPS) will be executed. The execution of a rule can change the status of the nodes, namely from active to inactive, or from inactive to active, via the **GoTo** action. When a node becomes inactive, its rules will be removed from the active rule list and all applicable rules belonging to that node, if they exist, will be also removed from the stack of applicable rules. Likewise, when an inactive node becomes an active node, all its rules will be put in the active rule list and the inference engine will check them for applicability. Aside from these effects, the execution of a rule can also change values of the system variables via the **Set** action. If there are value changes to system variables, the inference engine re-checks the related stand-by functions and active rules for applicable functions and rules. The system will continue such check-execution loops until there are no applicable functions and no applicable rules. Then, the questions in the active nodes will be asked in turn. However, after each input, the inference engine will repeat the check-execution loops and then ask for further input. This check-execution-query process continues until a conclusion is reached. The system can also be halted with the **Print** action, which outputs the interpretation or remarks, designated in the knowledge base by the knowledge base developer. The conclusion can be the output of the values of a series of variables or constants. Figure 11 gives the schematic presentations of the inference process.

Based on this rules cycling concept and the possible multi-trees-structured knowledge-based system, the system executions can be multithreaded at certain steps during a reasonning process, that is, rules form different active nodes become applicable and executed at same time regarding to the same system status. Therefore, one can formulating sub-trees doing certain procedural process (for example calculating the construction cost of different on-site infiltration measures) and related to variables (for example, a variable which holds always the currently selected measures in a reasonning process). In the main inference thread, these variables will trigger the processes represented in the relevant sub-trees in appropriate steps and the results of the sub-trees can be referenced further in the main inference process. In this way, the variables which trigger the sub-trees processes are similar to objects used in the CLIPS tool and the sub-trees processes are similar to their message-handlers.

A	Main flow of inference engine	2c	Check all applicable rules to delete these which does not belong to an active node any more
0	Update active nodes and hence active rules	2d	Re-check related applicable rules to delete these whose conditions are not satified any more
1	Check stand-by function for applicable function	3	Execute all applicable functions
2a	Check all active rules for applicable rules	4	Execute an applicable rule
2b	Check related active rules for application	5	Ask a question from active nodes

Fig.11a Explanations of each sub-flow chart

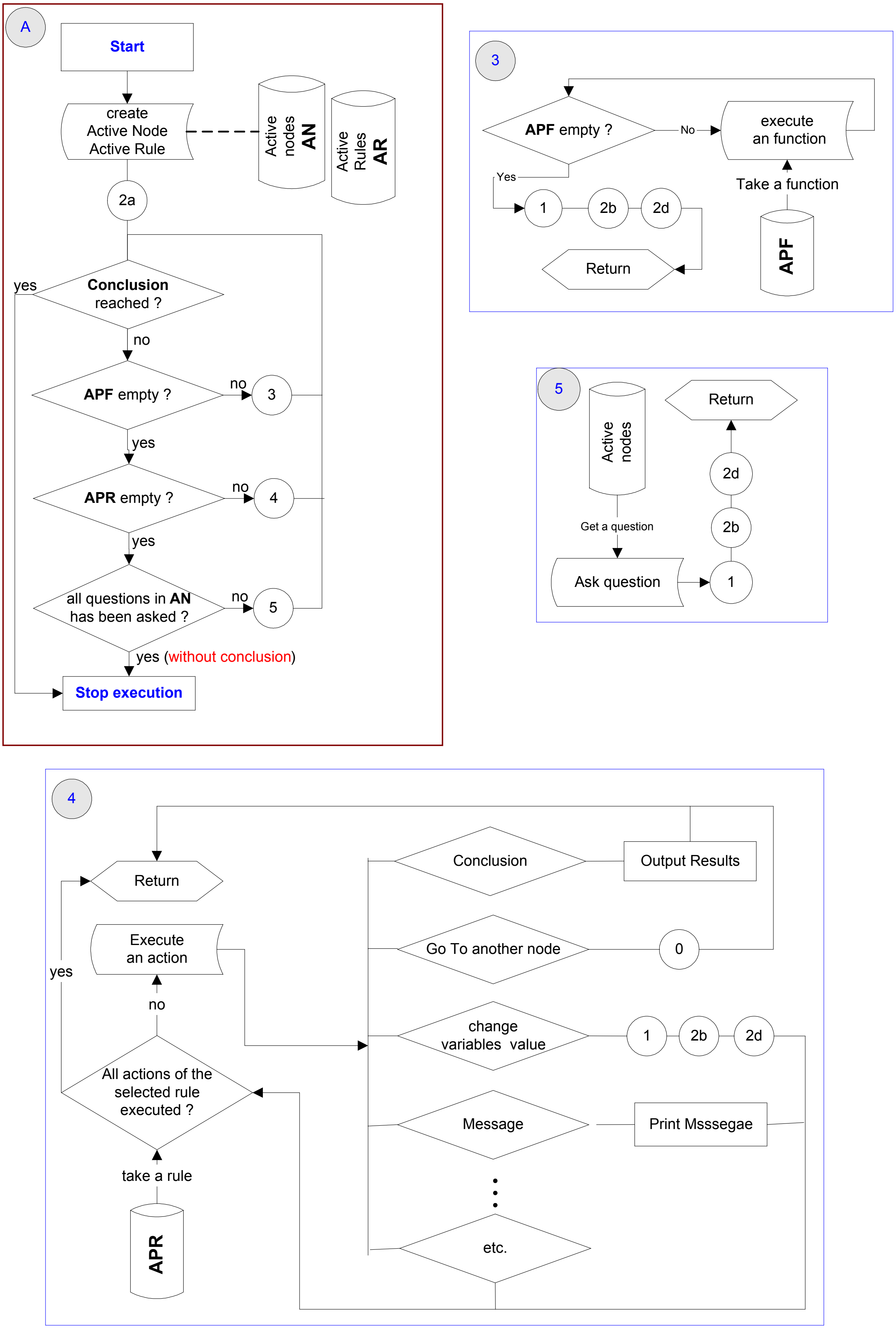


Fig.11b Schema of the inference mechanism in Flext (independent engine)

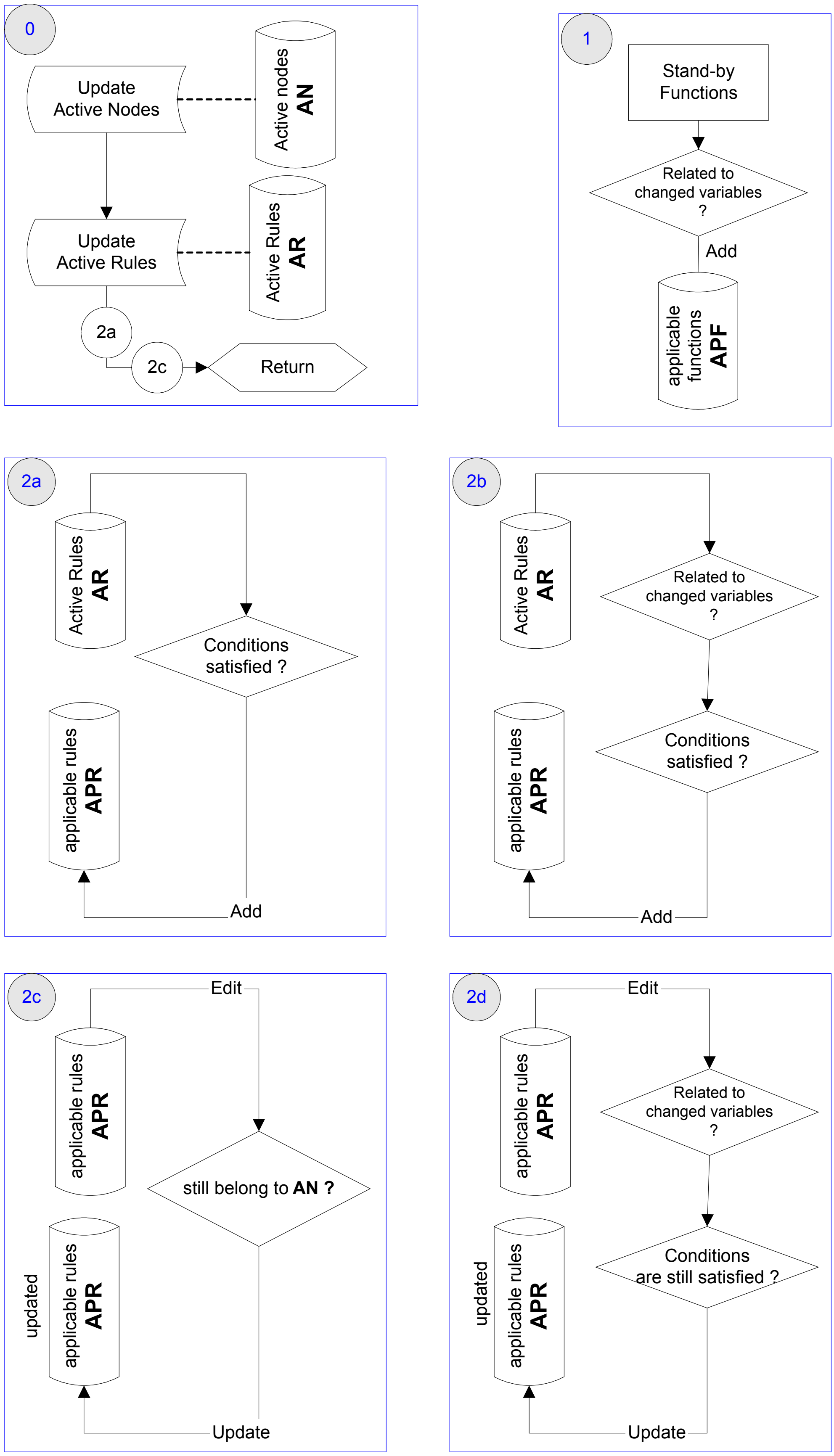


Fig.11c Schema of the inference mechanism in Flext (independent engine)

Data structure

As described in the introduction to expert systems, an inference engine of an expert system shell or tool should be general to any knowledge base in its own format. For Flext knowledge bases, the inference engine must be able to process nodes (each with different rules and questions), rules (each with different conditions and actions), stand-by functions (each with different operations and arguments) and questions. To make an inference engine apply to all these elements, whose structures change from node to node, from rule to rule, etc. both inside a knowledge base and between knowledge bases, the key point is to create regular or uniform storage structures for each of these elements, so that, although the conditions or actions, etc. may change from rule to rule, etc., their storage structures are always the same. This way, the inference engine can process them in a fixed logic and this logic can then be independent of the concrete rule conditions or actions, etc. and programmed as a general inference engine. In the following, some of the main data structures used for developing an independent inference engine are described.

A condition in a Flext knowledge base usually consists of a compound logic expression which includes further mathematic functions. All the operations in a condition can be dissected into single operations, for example: **(((x+y)^2)/(a+cos(b+c))>100) || (z<0)** can be executed sequentially as:

1): x + y
2): 1)^2
3): b + c
4): cos 3)
5): a + 4)
6): 2) / 5)
7): 6) >100
8): z < 0
9): 7) || 8)

These single operations can then be stored in a regular format as follows.

Table 4. Condition table

	operation	Argument 1	Argument 2	Temporary result
1	+	i_x	i_y	
2	^	1)	2	
3	+	i_b	i_c	
4	cos	3)		
5	+	i_a	4)	
6	/	2)	5)	
7	>	6)	100	
8	<	i_z	0	
9	||	7)	8)	

In the table, the argument1/ argument2 cells stores the addresses of the variables or the value of constants, that is, i_x, i_y, etc. stand for the address of the variable x, y, etc. in the variable table (tab. 6), 1), 2), etc. stand for the addresses of the relevant temporary results in this table. During the check process, the nine single operations will be carried out sequentially and the ninth temporary result will be the final result of the condition, i.e. true or false.

In this way, conditions of all rules in a knowledge base are dissected and stored in the condition-table. The address of the start operation and the address of the end operation of a rule condition in this table are referenced in the rule-table (tab. 7).

Similarly, a function table is constructed for storage for the stand-by functions and the functions used in the **Set** action of rules. The first operation address and the end operation address of each function are referenced accordingly in the stand-by function index-table or the action-table (tab.5). The function table is nearly identical to the condition-table. The only difference is that the final result of a function can be any data type. They are separately stored in different tables (corresponding multi-dimension arrays in codes) in consideration of:

1) the clarity of the codes,
2) avoiding possible search a long address (> 32767), if a knowledge base is large enough.

Actions of a rule can be consists of many sentences, each with a keyword, so the action syntax must be first broken down to a set of single actions. The information to be stored in an action table depends on the keyword of each action. For example, the following actions can be dissected as in Table 5.

Set x=(a+((b*c)/d)), y=88
Print "Temporary Info:","a=",a
Conclusion "This is demo", " X=", x

Table 5. Action table

Keyword	Field1	Field2	Field3
Set	i_x	Start addr. In function table	End addr. In function able
Set	i_y	88	
Print	Start-addr. in print table	End-addr. In print table	
Conclusion	Start-addr. in conclusion table	End-addr. In conclusion table	

In this table, the **Set** action is further dissected into two actions, i.e. setting variables *x* und *y* with a constant and a function respectively. So, there are in total four single actions in this example, which can then be stored in four neighbouring rows in the action table. The first column stores the keyword. The information stored in the other three columns depends on the concrete action. For example, the first row stands for setting the variable *x* with a function, which is further dissected and stored in the function table. The addresses of the start and end of this function in the function table are referenced here in column 3 and 4 respectively. The address of the variable *x* in the variable table is stored in column 1. In contrast, the second row stands for setting a variable with constant, therefore the constant *88* is directly stored in column 2. The third row stands for the message printed during execution. All printed messages in a knowledge base are stored in a print-table. So, the addresses of the start and end of this **Print** action in the print-table are referenced in column 3 and 4 respectively. The process of the **Conclusion** action is similar to the **Print** action.

All variables used in the knowledge base are stored sequentially in following uniform format.

Table 6. Variable table

Name	Value	Related rules	Related functions
X		R_i, R_j,…	F_i,..
Y		R_k,…	F_j, F_k,..

Ri, R_j, etc. means the address of the related rule in the constructed rule table; F_i, F_j, etc. means the address of the related stand-by functions in the constructed function index table.

The related rules of a variable indicate rules in whose conditions the indexed variables are present. Similarly, the related functions indicate functions in whose expressions the indexed variables are present. These indexes are made to help the inference engine to check only related rules or stand-by functions for applications during runtime if a certain indexed variable changes its value.

In the following table, rule information (the start and end addresses of their conditions in the condition table, the start and end addresses of their actions in the action table, the address of their nodes in the node table) are stored.

Table 7**.** Rule table

Rule	Node	Start	End	Start	End
1	Addr. in node table	Addr. In condition table	Addr. In condition table	Addr. In action table	Addr. In action table
2					
3					
4					

4.3.5 Edit modes in knowledge base formulator

The Flext tool is characterised not only by its knowledge representation (simple rule formulation syntax and the structuring of the rules system, question system), but also by its convenient knowledge acquisition system, the knowledge base formulator.

The knowledge base formulator has two edit modes: table mode and graphic mode. These are shown in figures 12 and 13. In the table edit mode, rules in a node can be written sequentially in a table. Each table corresponding to the whole screen is for the definition of rules and questions in one node. By clicking the command buttons **Up** and **Down** to the right of the rules definition fields or the question definition fields, rules or questions can be scrolled on the screen. In one node, as many rules as necessary can be formulated. Similarly, as many questions as necessary can be specified in one node. With the command buttons **Previous** and **Next**, one can navigate within a knowledge base from node to node. By clicking buttons **Insert Before**, **Insert After** or **Delete**, one can add and delete nodes. The sequence of rules in a node corresponds to their sequence in its edit table while the sequence of nodes in a knowledge base corresponds to the sequence of their edit tables in this table-editting mode.

In the graphic mode, the logic relationship between nodes is clearly presented. Nearly all the edit functions as in table mode are also available in graphic edit mode. In this mode one can better manipulate the rules or nodes like delete or insert because it is clear where in the decsition trees one has delete or insert a rule or a node. Similarly a node can be clearly redirected from one node to another node. In this sense, correcting or modifying an existing knowledge base is much easier than doing it in table edit mode or in a text editor like in CLIPS. In addition, the graphic mode provides the possibility of visually tracing a reasoning process during an execution of the formulated knowledge base. This is especially helpful for testing a knowledge base. It is also the premise for the participation of a user or decisionmaker into the decision process, if this tool is used for building a knowledge-based decision support system.

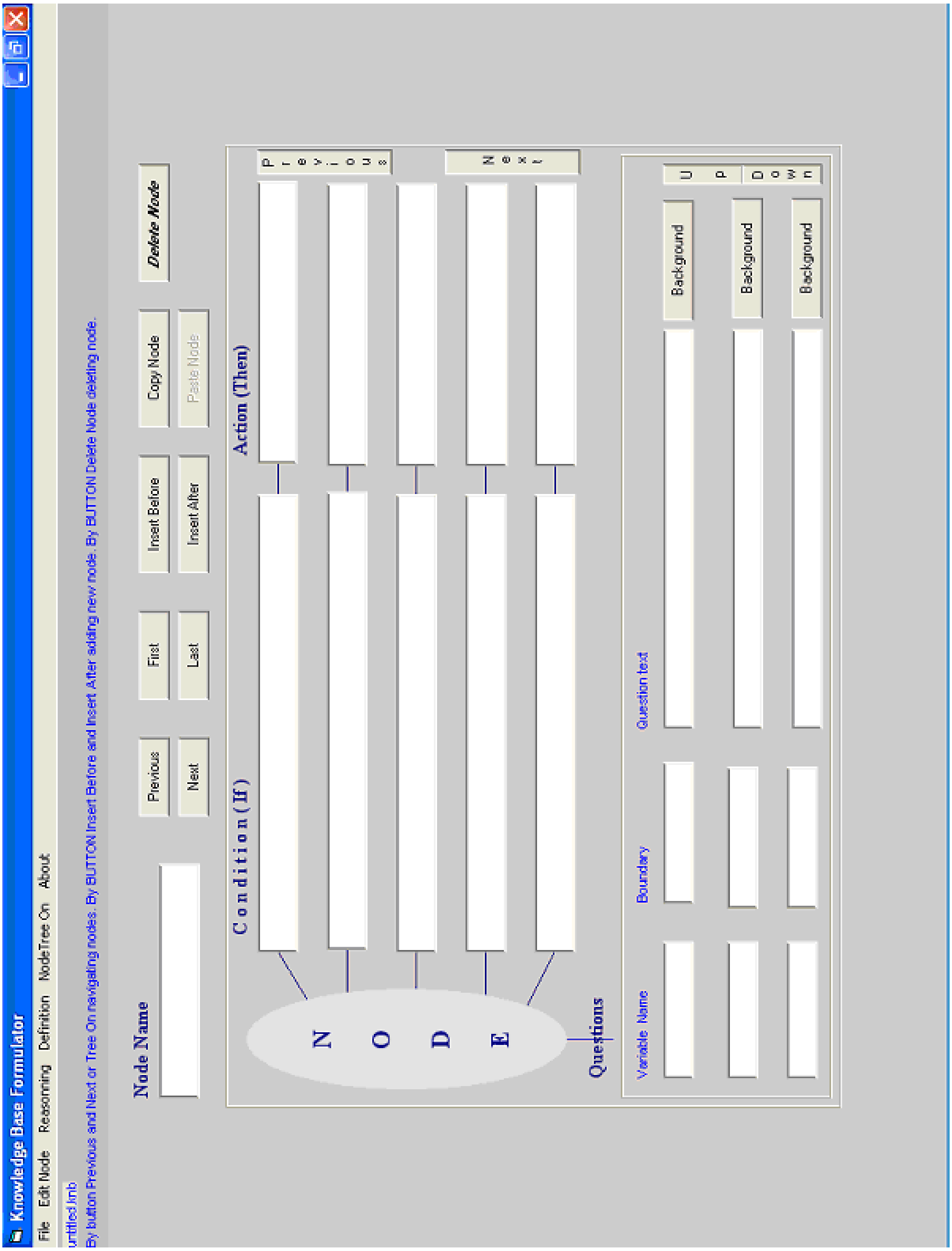


Fig. 12 Table edit mode of Flext knowledge base editor

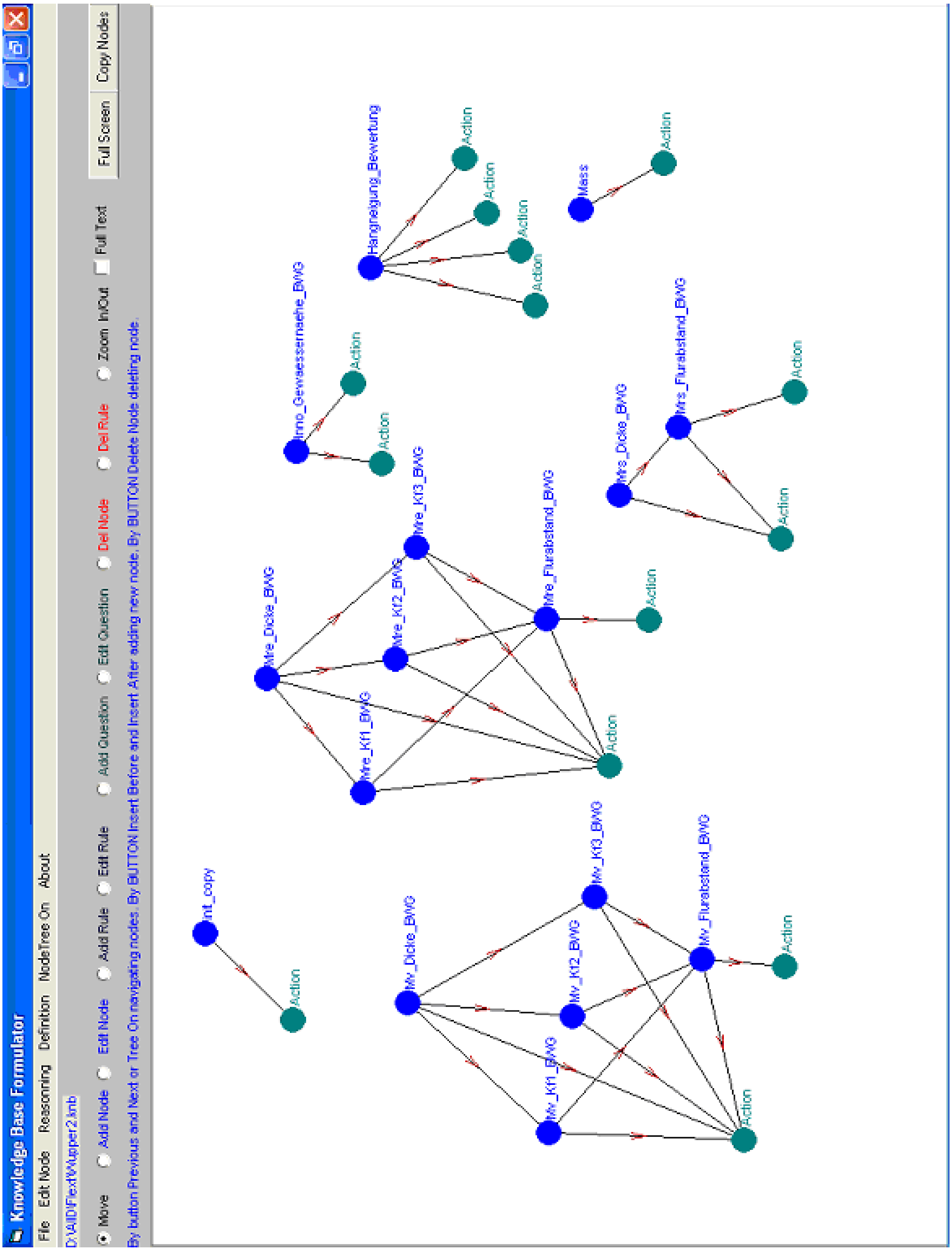


Fig. 13 Graphic edit mode of Flext knowledge base editor

In addition to editing rules and questions in a knowledge base, the formulator also provides an interface for matching questions and output variables to the fields of a designated database record set. This is used for the automated repeated execution of an ES based on the selected database record set. In each execution, the necessary inputs will be automatically read from pre-matched fields of the current processed records. Outputs will also automatically be written to pre-matched fields of the same record from which the inputs are read.

The table edit mode provides a clear overview of the contents of rules and questions side by side in a node, while the graphical mode enables the clear overview of the inference logic of the whole knowledge base. Chapter 7 describes the use of the knowledge base formulator in more detail.

4.3.6 Characteristics of this tool

The Flext tool is simple and flexible. Its main properties are listed as follows.

1) It is adequate and flexible enough to represent the desired knowledge for the specific problem in this study,
2) It has an easily understood representation formalism (natural language for syntax, explicit logical rule structure in trees or network,
3) It supports question-specific graphical interfaces,
4) It provides an opportunity to debug the formulated knowledge base,
5) It has a high-level knowledge acquisition interface,
6) It is integratable into other applications, for example, a GIS platform;
7) It is able to integrate external applications, for example, a professional simulation model;
8) It is able to manipulate database operations, such as an SQL search;
9) It is convenient to input from or output to external data files, such as text files, databases, etc.

In summary, the Flext tool has a simple knowledge representation mechanism (conventional syntax, structural rules system, automatic question processing system) and an advanced knowledge acquisition system (i.e., table and graphic edit interface).

4.4 Integration of Flext into GIS

4.4.1 Concept of integration of an ES into GIS

In the sample application of this study, GeoMedia Professional (a GIS product of Intergraph Inc.) is selected as the GIS platform for integrations.

Geomedia Professional enables users to install their own utilities developed in Visual Basic, etc. as so-called user commands. It also provides dynamic libraries with a large amount of functions on spatial operations or computations and database operations for users to develop their commands. In such commands, relevant procedures can be written to capture events such as a mouse click on an object of a feature, etc.. In reaction, the user commands may execute its programmed process and return the relevant results from the executed process. In this way, the developed ES tool Flext can be integrated into Geomedia Professional; that is, embedding its functions in a user command source code. Hence, any problem-specific expert system (i.e. loading Flext with a specific knowledge base) can be integrated into Geomedia Professional.

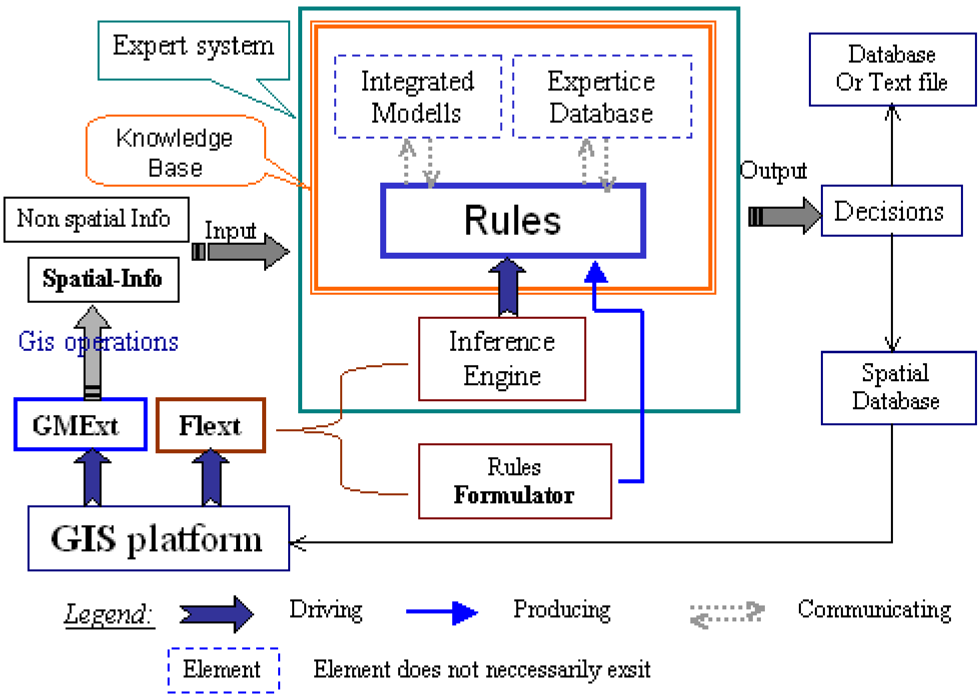


Fig.14 Scheme of integration of Flext into GIS platform

In this study, in addition to Flext, a GIS extension programme GMExt for pre-processing the spatial data for the expert system is also integrated into Geomedia Professional through a user command. Figure 14 shows the schema of the integration of Flext and GMExt into a GIS platform.

4.4.2 Running procedure of the integrated model

One can now easily start the knowledge base formulator from Geomedia Professional to edit one's problem-specific knowledge base, which usually consists of a rules system and maybe external professional models and expertise databases referenced in the rules system. Users must prepare their necessary external models either by programming them in conventional programming language like Visual C++ or Visual Basic, or by obtaining them from other sources. The necessary expertise database can be prepared by using MS Access.

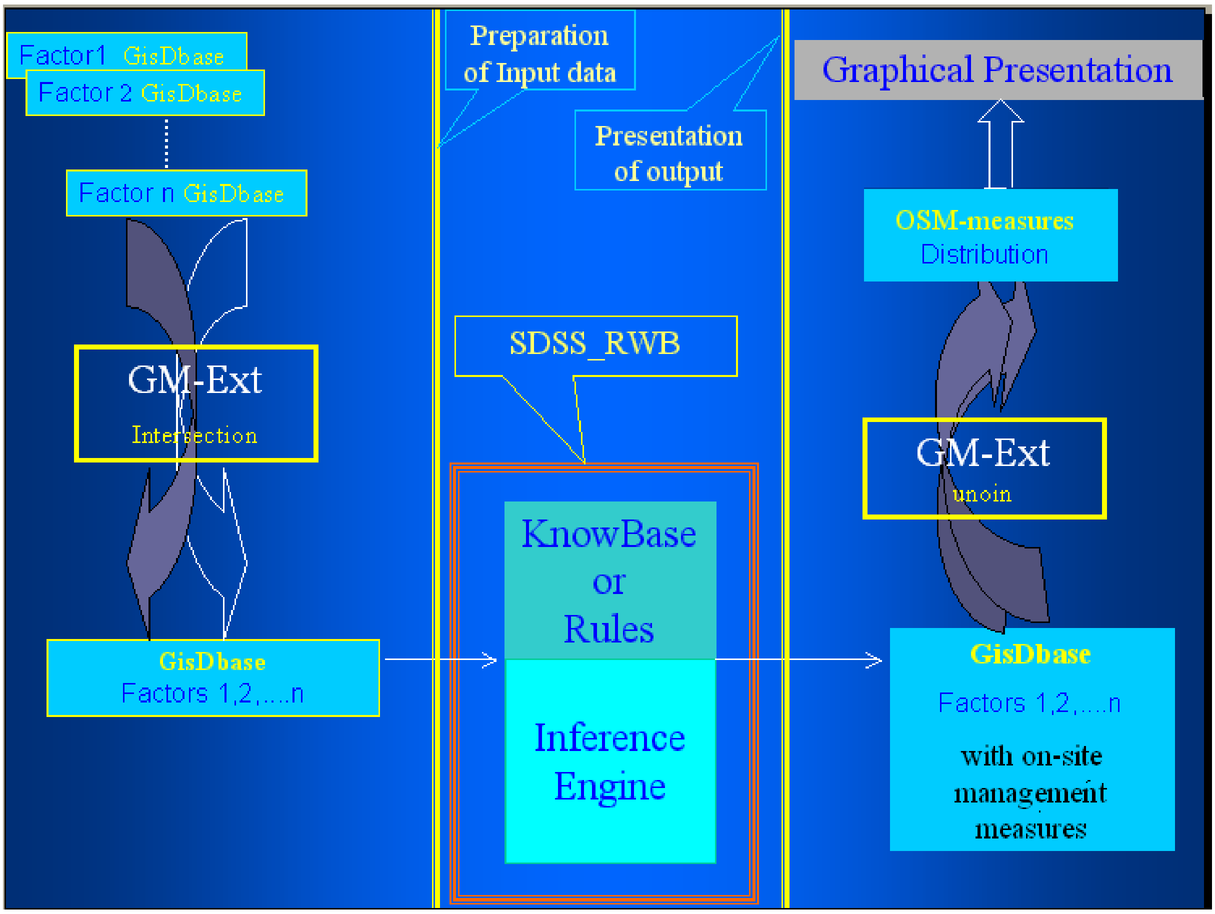


Fig. 15 Schematic presentation of the application of a GIS integrated expert system for the planning of on-site stormwater infiltration

The formulated knowledge base and the Flext Runtime modular (inference engine and interface) form a problem-specific expert system or knowledge-based decision support system. This system can be activated through menu commands of Geomedia Professional (user commands) to process for all objects of a selected feature. The runtime module can be also triggered by clicking on an object of a spatially distributed feature. For example, in reacting to a mouse click, the integrated knowledge-based system is executed with the necessary inputs read from the pre-matched fields of the record related to the clicked object. The results or decisions for this object are then displayed in a docking window.

For the planning tasks described in chapter one, the procedures can be schematically demonstrated as in Figure 15. Chapter 5 gives more details in this regard.

5. Development of transparent knowledge-based spatial decision support systems for on-site stormwater management planning in urban areas

For the development of a transparent spatial decision support system for the planning of on-site stormwater infiltration measures, the necessary case-specific knowledge was first analysed and clearly formulated in natural contexts. The transformation of human knowledge into a knowledge base for Flext and the application of the developed system was then demonstrated in detail. There are two knowledge bases constructed in this case study. Both systems are for planning decentralised stormwater management. Therefore, before the concrete case-specifc analysis, a general description about on-site stormwater management is given in following section.

Of decentralised rainwater management measures, the on-site rainwater infiltration measures, which serve as making good use of the water storage capacity of local soil, are of special interest in this study. Among on-site rainwater infiltration measures, some are designed such that the runoff from the connected catchment (local sealed area) can be totally infiltrated underground and later either percolated further into groundwater bodies or evapotranspirated before the next rain event. However, these so-called complete infiltration measures are sometimes not justified in soil with very low permeability. In such situations, measures aimed at partial infiltration are then considered.

5.1 On-site stormwater infiltration measures and their influencing factors

5.1.1 On-site stormwater infiltration measures

<u>Surface infiltration</u>

One surface infiltration system is the so-called pervious paving system. It is usually applied for streets with very low transport loads, yards or parking areas. Instead of complete sealed paving in these areas, pervious paving can be achieved either through elimination of the finer aggregates in raw paving material to create gaps in a continuous asphalt or concrete surface course or by leaving uniform gaps between individually sealed concrete paving blocks (pavers) when the pavers are used for surface paving (Fig.16). In the later case, the gaps can also be created by using perforated pavers. For both pervious paving systems, a water storage bed of crushed stones can be installed under the surface course, if necessary. In addition, short grass is usually planted in the uniformly distributed gaps in the paver surface course.

.

A pervious paving system not only reduces stormwater runoff but also reduces or retains total suspended solids and part of the pollution load in the stormwater runoff due to its filter function.

However, whether pervious paving is appropriate to a given area depends mainly on the following critical factors:

- pollution loads
  To avoid the danger of groundwater contamination, areas with a possible high pollution load, such as industrial or commercial areas, areas near gas stations or vehicle maintainace facilities, etc., are not appropriate for pervious paving systems.
- transport loads
  Due to the reduced shear strength of the paved surface course, pervious paving is limited to areas of relatively infrequent use by light vehicles.
- potential of damage caused by seepage
  To avoid basement seepage, flooding or damaging other subsurface structures by seepage, pervious paving is not justified in areas where the groundwater table is high.
- permeability of the subsoil
  To achieve reasonable infiltration, pervious paving is only useful if the permeability of the local subsoil is greater than $2x10^{-5}$ m/s. To design a pervious paving system, it is normally assumed that no significant water accumulation will occur on the surface for design storm runoff.

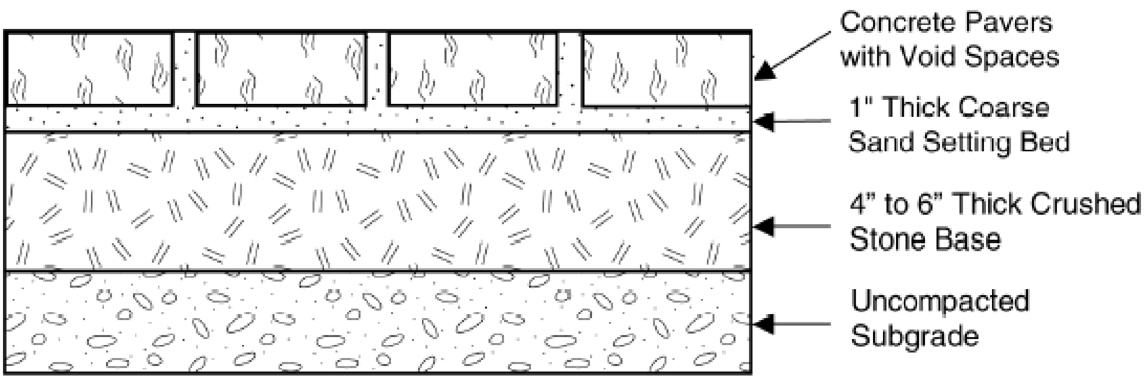


Fig.16 Permeable paver without storage base
(source: New Jersey Stormwater best management practices manual 2004)

Another surface infiltration system is for the storm runoff from long distance transportation roads or highways, along which pervious vegetated depressed strips usually exist on one or both sides. These bushy or grassy strips can be designed as a surface infiltration system for the runoff drained from the roads. However, considering the heavy transportation load on these roadways, it is doubtful whether the constriants on limited pollution load are met.

In general, surface infiltration best approches the infiltration style of the natural area.

Swale/Basin Infiltration

The simplest infiltration swale is a shallow vegetated depression in a private or public natural area. However, the swale can also be a man-made rectangular basin with a relatively deep infiltration surface. in a broad sense, the central or semi-central infiltration basin situated at the outlet of the sewer system is an infiltration swale. They are usually arranged in cascades. The sewerage overflow is led into the basin cascades sequentially and infiltrates finally completely into the subsurface. An important characteristic is that the sewerage overflow must pass through a vegetated surface, through which the pollutants can be permanently eliminated, according to Schneider (1999), Sieker. F, etc. (2001), Sommer. H (2002). Another important property is that lasting water accumulation in the basins is not allowed. The permeability of the bed layer and the storage volume of the basin should be designed such that the time of water logging in the infiltration basin is less than 3 hours, which ensures the lasting infiltration capacity through enough ventilation of the basin bed. The constraints for the installation of an infiltration basin are:

- The permeability of the basin bed layer must always be greater than 5×10^{-6} m/s.
- The groundwater depth must be large enough so that the infiltrated water can be constantly drained away and no perched groundwater occurs, and hence no risk of any damage caused by seepage.
- A natural area of about 10% of the planned connected sealed area for the installation of the infiltration swale should be available. This criterium is actually related to the local rainfall statistics and the design standard. The objective is to guarantee that the maximum water depth in a basin is no larger than 30 cm (in Germany) or 2 feet (in the USA), etc.. For example, for an average rainfall event in Germany, if the infiltration basin is designed to achieve the standard of the frequency of overflow once every five years, the necessary effective basin volume should be no less than 20 liters per square meter connected impervious area. Therefore, if the available natural area for installing a basin or, in other words, the basin bottom area amounts to about 10 percent of the connected impervious

area, then the necessary effective swale depth is about 20 cm, which is under the maximum allowed water depth in basin (30 cm).

- As described in installation of pervious paving systems, areas with expected high pollution loads are not appropriate for the installation of an infiltration basin.

.

Swale-trench infiltration

For the area where the infiltration swale is not justified due to the very low permeability (i.e. less than $5x10^{-6}$ m/s)) of the local soil, the so-called infiltration swale-trench system should be considered. However, based on practical experience, the permeability of the local soil should be greater than $1x10^{-6}$ m/s for using a swale-trench system.

As shown in figure 17, the collected runoff is first led to a vegetated surface swale. From the swale, the rainwater is infiltrated and percolated through an active soil layer into an artificial trench of gravel, etc.. The trench should be constructed so that the water stored in the trench can be percolated from all sides into the surrounding soil. The effective storage volume of the trench of gravel, etc. amounts to 25-40% of the total trench volume. However, the effective storage of a trench can be significantly higher (greater than 90% of the total trench volume) if it is constructed of artificial three-dimensional grid shaped material. The geo-textile is necessary to cover the trench to prevent it from being filled with fine material.

The total storage of the infiltration swale-trench should be around 50 litres per square meter of connected sealed area, which should further be distributed as 20 litres in the swale and 30 litres in the trench, respectively. There should also be a shortcut path for directly leading the possible overflow from swale to trench. Similar to the requirement in an infiltration swale, the groundwater table should always be under the bottom of the trench so that the infiltrated water can percolate further into the surrounding soil without causing seepage damage on adjacent basements and subsurface structures.

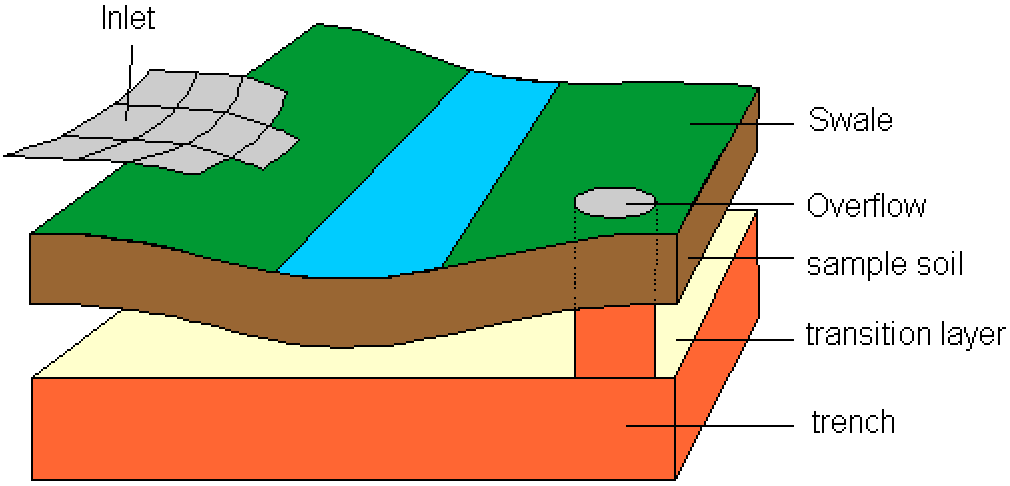


Fig. 17 The schematic presentation of a infiltration swale-trench element

The constraints to the installation of an infiltration swale-trench system are the same as that for an infiltration swale, except that the latter requires higher permeability of local soil.

Well infiltration

The infiltration well is the oldest technology in dealing with rainwater infiltration. It usually is designed as a well of several meters in height, with the bottom significantly above the groundwater table. The effective storage capacity between the bottom and the elevation of the inlet should be designed as with infiltration swales and infiltration swale-trench systems at 20 to 50 litre per square meter of connected sealed area. Due to the comparatively small infiltration area (around 1/10 of infiltration swale), the height of water column per unit area in the infiltration well and the corresponding pollutant load per unit area are significantly higher than with infiltration swales or swale-trench systems. The high pollutant load leads to pollution of the surrounding soil and groundwater. For this reason, the infiltration well is nowadays not recommended for urban stormwater management.

Pipe-trench infiltration

In a pipe-trench infiltration system, the rainwater is directly led into an underground pipe which is constructed with gravel and artificial grid material. The water is then temporarily stored in the pipe and at the same time percolates slowly into the surrounding soil. The advantage of such a system is that no natural surface area is needed to infiltrate and/or store rainwater. However, due to the risk of potential pollution of the surrounding soil and groundwater, this method is nowadays not recommended for stormwater management.

Swale-trench-infiltration-drainage-system

The above-mentioned measures for the infiltration of rainwater are all designed to completely infiltrate the rainwater collected from the connected sealed area. They are the opposite of the sewer system in the sense that the sewer system drains rainwater completely away from where the runoff is produced. In reality, both methods have their advantages and disadvantages and limitations in practice. For example, complete infiltration requires a high soil infiltration capacity, a characteristic which is sometimes not the case. An idea has been developed to combine the advantages of both concepts, that is, make good use of the available infiltration capacity through the construction of decentralised infiltration measures, and draining away the rest of the water through connection of the infiltration system to the sewer system. The infiltration swale-trench-drainage-system (Mulden-Rigolen-System or MR-System, in Germany) is described as following.

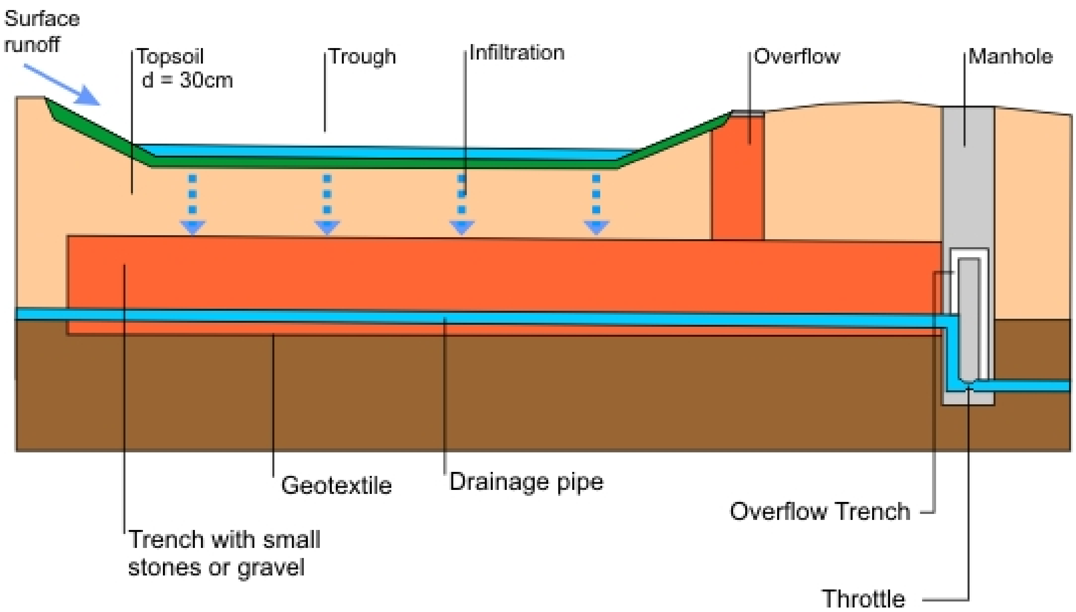


Fig. 18 Swale-trench-infiltration-system with throttled drainage (H. Sieker *et al*, 2002)

General Principles

The MR-System combines three elementary engineering techniques to manage rainfall runoff: infiltration, storage and drainage by sewers. The principal considerations of the MR-System are:

- Infiltration as much as possible, depending on the permeability of the soil und the infiltration area available
- Storage as much as necessary, to support and prolong infiltration processes
- Drainage by throttled sewers as final means to ensure the drainage standard demanded

A further principle of the concept is to decentralise the measures as much as possible in order to spread groundwater and minimize the costs of the sewer system. This means measures must be located near impervious elements such as roofs, parking lots, roads etc.

Another basic requirement is that infiltration must occur through the active upper soil layer, which is covered with grass or other suitable vegetation. This way, the stormwater quality is enhanced by partial removal of solids (with adsorbed chemicals) and dissolved chemicals.

In general, the aim of the MR-System concept is to bring the equation of water balance of urban areas closer to its natural state before the areas were developed.

.

Description of the standard element

Fig. 19 and fig. 20 show exemplarily a cross and a longitudinal section of a standard element of the system.

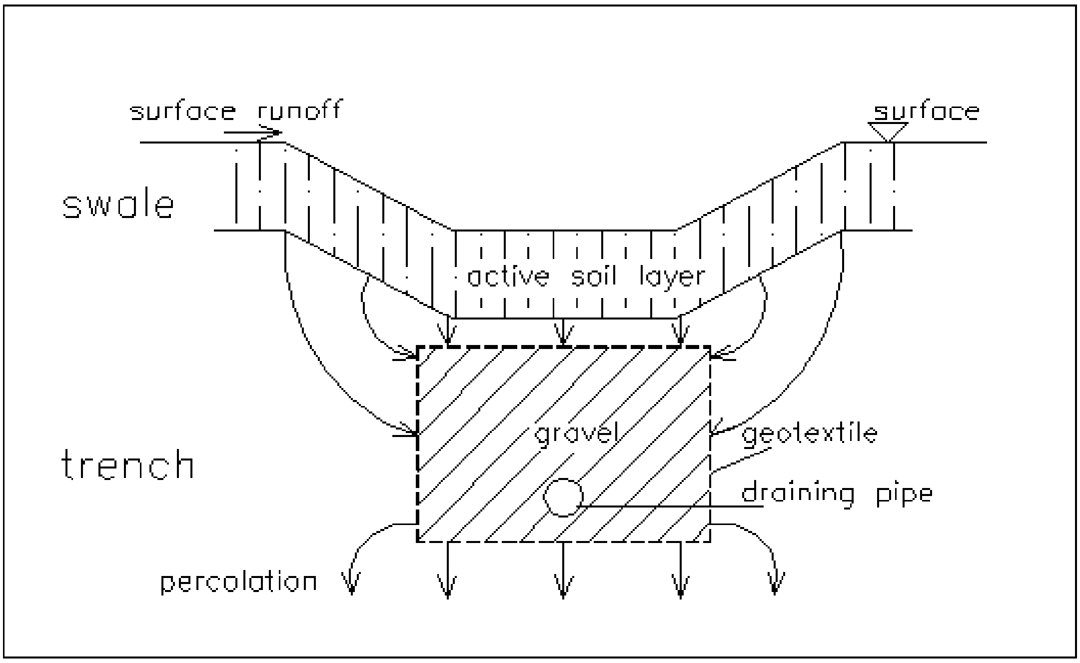


Fig. 19 : Cross Section of the MR-Element (source: F. Sieker, 2000)

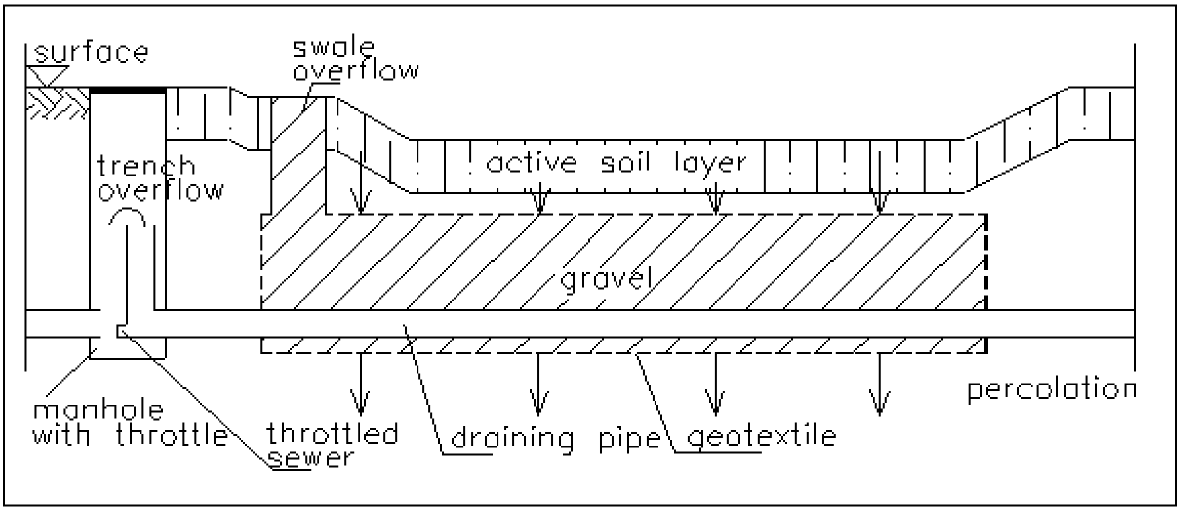


Fig. 20: Longitudinal Section of the MR-Element (source: F. Sieker, 2000)

The stormwater runoff is discharged into the shallow, grassy swale by gutters etc. on the surface in order to avoid deep swales (which would be the case using underground feeder pipes). The depth should be no more than 0.3 m. The minimum permeability value of the active soil layer must be 10^{-5} m/s to ensure that the infiltration process will be finished about 3 hours after the end of a rainfall event. According to experiences in Germany, the storage volume of the swale can be estimated by assuming a runoff of 20 mm from the connected impervious area. The banks of the swales should have a slope of about 1:3 on behalf of maintenance.

The infiltration trench under the swale is designed as a conventional underground infiltration trench, formed by an prismatic body of highly porous material such as gravel or lava granules, wrapped in a permeable geotextile. This trench provides long-term storage of stormwater, as opposed to the short-term storage in the swale. During typical events, the trench is fed by infiltration through the bottom of the swale only, but in the case of heavy rainfall, swales may overflow directly into the trenches. For this, a swale-overflow is implemented as a short-circuit between the swale and the trench. The swale-overflow can be implemented in several ways, e.g. as a gravel-filled pipe, covered at the top by a protective hood. Between the swale-bottom and the top of the trench, an intermediate layer of sand is installed. The pre-treated stormwater percolates from the trench into the surrounding soil. In case of temporarily elevated groundwater tables or excessive soil moisture, the trench can be used not just to manage the infiltrated stormwater runoff, but also to manage undesirably high groundwater or soil moisture. The demanded storage volume of the trench, or the usable pore volume of gravel etc., can be estimated within the first design step by assuming another 30 mm runoff from the connected impervious area.

In the lower part of the trench, there is a drainage pipe which serves as a bottom outlet and can be used to drain the trench completely, if necessary. This bottom outlet discharges into a manhole, in which the stormwater outflow can be throttled to the permissible discharge. Finally, a trench overflow device ensures that the trench is filled only to its top edge.

The above-described swale-trench-elements are located both on private and public sites, even along traffic banks. Several such elements can be linked by throttled sewers to a MR-System, using either parallel or in-series arrangements. The final outflow from such a sewer system is designed for a permissible discharge that is comparable to the drainage area discharge before urbanization, according to the zero-runoff-increase concept.

The MR-System is almost universally applicable, independent from the permeability of local soils and other circumstances. The only possible constraints on its applicability may be steep surface slopes of the drainage area or a lack of space in densely developed downtown areas. The layout of the MR-System is rather flexible and can be easily adapted to various soil conditions. In soils with intermediate permeability (kf 10^{-6} m/s), where no discharge from the trench is expected, the sewer system can be eliminated and the MR-System just comprises unlinked swale-trench-elements, known as the unlinked MR-System. In soils with high permeability (kf 10^{-5} m/s), even the trenches can be eliminated and the system is reduced to swales only, that is, the unlinked M- System. Both these variants are special cases of the overall MR-System. In practice, a certain urban area may be divided into parts of different soil conditions so that the application of different variants may be required.

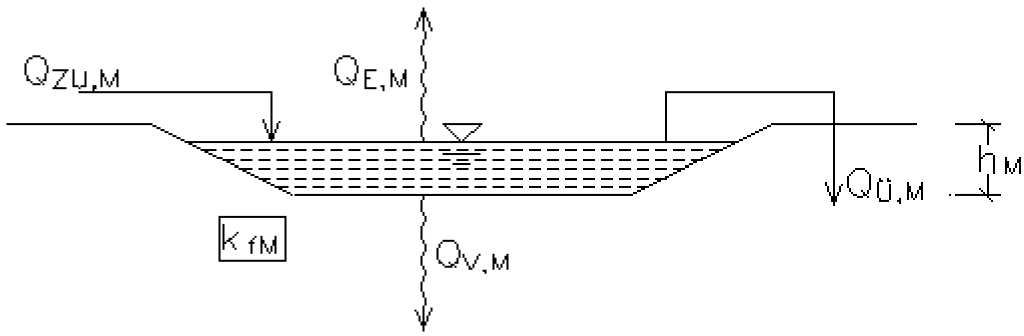


Fig. 21: Components of the Swale (source: F. Sieker, 2000)

Computations

After pre-design of the MR-elements by the assumptions of storage volumes mentioned above, it must be proven that a given recurrence interval of failure will be met. In Germany, a recurrence interval of 5 years is standard. It must be proven by a so-called continuous long-term simulation of the rainfall-storage-infiltration-outflow process that the pre-designed element will meet that requirement. Continuous means that the input series of rainfall data must include dry periods.

Fig. 21 shows the components of the swale, which must be taken into account:

- $Q_{ZU,M}$ = inflow from the connected area (roofs, roads etc.)
- $Q_{E,M}$ = evaporation and transpiration during and after rainfall events
- $Q_{Ü,M}$ = swale-overflow into the trench, functions only if the swale is overloaded
- $Q_{V,M}$ = percolation through the bottom of the swale either into the natural soil or into the trench
- $k_{f,M}$ = permeability value of the active soil layer of the swale (kf 10^{-5} m/s)
- h_M = depth of the swale (in general: 0,3 m)

As a rule, the computations are normally carried out using time steps of 5 minutes in order to take into account the high fluctuations of the inflow data. A continuous series of rainfall and dry period data of at least 10 years is used. As a result of the inflow-outflow-equation, the number of events exceeding the given depth h_M will be obtained. If the number of years of the input series divided by the number of failures exceeds the given recurrence interval, the design of the element must be extended or vice versa.

In case of a trench below the swale, the working-frequency of the swale overflow is a free parameter to change the division of the whole storage volume of the MR-element between the swale and the trench. That may be necessary in the case of densely developed areas, e.g. along roads. Nevertheless, a working-frequency of five times per year should not be exceeded in order to force most of the runoff to be treated by passing through the swale's active soil layer.

Fig. 22 shows the inflow and outflow components of the trench with $Q_{Ü,M}$ and $Q_{V,M}$, the output components of the swale mentioned above:

$Q_{Z,R}$ = inflow from a trench above in case of a linked MR-System

$Q_{V,R}$ = percolation into the surrounding natural soil

$Q_{D,R}$ = throttled outflow from the trench (a permissible discharge)
$Q_{Ü,R}$ = trench overflow in case of filling to its top edge
h_R = depth of the trench

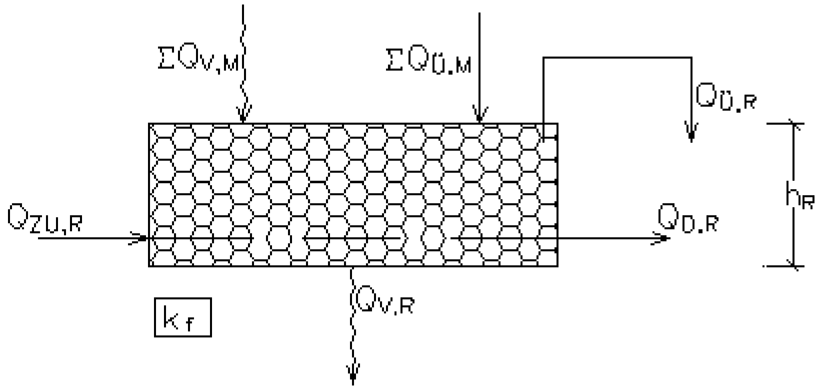


Fig. 22: Components of the Trench (source:F. Sieker, 2000)

The application of the inflow-outflow-equation to the trench in combination with the swale components aims to prove that a situation where both the trench and swale are filled to its top edge does not happen more than once within the given recurrance interval. If this is not the case, the designed storage volume of the trench or the swale must be changed.

A parameter that affects the proportion between the components $Q_{V,R}$ and $Q_{D,R}$ or $Q_{Ü,R}$ is the level of the trench outlet in relation to the bottom of the trench. If the outlet is raised in relation to the bottom of the trench the component $Q_{V,R}$ will increase and vice versa.

The whole procedure to compute MR-elements or MR-Systems can be carried out relatively easily by computer programs which are available and increasingly being applied in Germany

5.1.2 Classification of Influencing factors of decentralised rainwater management

Although some main constraints of certain on-site infiltration structures already described in the last section, factors concerning an overall decision or selection of a stormwater infiltration measure for a certain area are classified and explained in detail below.

1) factors affecting on-site water retention

On-site water retention is the essential pre-condition for decentralised stormwater management. Therefore, these factors will be investigated and evaluated first. The analysis of these factors for the whole catchment or planning area yields the so-called water-retention potential map, which can guide planners to narrow their planning to areas of infiltration interest. For these areas further work to obtain better raw data (such as data with more spatial resolution) for selecting the optimal type of on-site stormwater infiltration measure may probably be necessary. Water retention means, in this case, immediate on-site rainwater infiltration or infiltration after temporary storage. It is principally influenced by the following factors:

- Permeability (K_f) of the surrounding soil:a parameter indicates how fast the infiltrated water will be transported underground. The permeability mainly depends on the size of pores and their distribution in the soil. It is also a function of soil water content. In fact, due to the heterogenity of the soil structure, the permeability also varies spatially. Because on-site infiltration normally involves a relatively large amount of water that should be stored temporarily in the surrounding soil, the average saturated soil permeability is most relevant to on-site infiltration and is considered in the evaluation of on-site infiltration measures.

  The following table shows the K_f criteria on each infiltration measure according to German practices.

Table 8. Criteria of permeability on different on-site stormwater infiltration measures

Decentralised rainwater infiltration measures	Permeability (m/s)
Surface infiltration	$K_f \geq 2*10^{-5}$
Swale infiltration	$K_f \geq 5*10^{-6}$ Can be smaller when large pervious area available
Swale-trench element without drainage	$K_f \geq 1*10^{-6}$
Swale-trench-system with throttled drainage	No limitation

- Porosity (n) of the soil layer: indicates the extent of the space within the soil. The effective porosity reflects the storage capacity of the soil. The higher the effective soil porosity, the greater the storage capacity of the soil, provided the same water content in the soil at the beginning of a rain event. For on-site water retention potential, in addition to the soil porosity, the depth of the unsaturated zone or the distance the infiltrated water can be transported during a rain event, which depends on the soil permeability, are also influential. Unfortunately, the soil porosity usually varies indirectly with the soil permeability. The finer the soil material, the greater the soil porosity, but the lower the soil permeability. Therefore, in the evaluation, the soil porosity is eventually not considered as a constraint factor. After all, the water storage capacity can temporarily be partially supplemented with artificial storage in structures, e.g., the effective volume of an infiltration swale ('effective' to stress that the water depth in the swale is limited by its allowed maximum design depth), the storage volume in a trench (swale-trench infiltration).

- Groundwater table: reflects the thickness of the unsaturated layer, and hence the local water retention capacity, and therefore is essential to on-site infiltration measures. Moreover, if the groundwater table is shallow, there is a risk of the damage to basements or other shallow subsurface structures due to seepage caused by the rise of the groundwater table due to infiltration.
- Soil thickness: indicates the distance to solid rock. Although solid rock could also have a significant permeability and porosity, this is usually unstable and difficult to determine, so on-site infiltration measures should be avoided in areas where the thickness of the soil or uncompacted sediments is low.

2) Factors concerning pollution dispersion
The infiltration of collected rainwater will inevitably transport pollutants underground, even though there are vegetated filtering processes in the swale and the active soil layer at the bottom of the swale, so infiltration measures should be avoided for such environments as:

  - well zones where groundwater is pumped for drinking water, because the demand on water quality is very high.
  - highly polluted areas such as old industrial areas, industry deposits, roadways with high traffic volumes, etc., because high pollution loads are expected in the stormwater runoff from such an area.

3) Factors affecting the construction of on-site infiltration measures
  - Surface slope concerns both the installation of the infiltration basin and the inlet of the stormwater runoff from the connected sealed area to the basin. The steeper the slope, the higher the cost of constructing the infiltration basin.

- Thickness of soil layers: described already in the first class, this also influences the constructon of infiltration structures, for example, the underground trenches.

- Available natural area: for swale and swale-trench systems, an existing natural area (at least 10 percent of the planned connected sealed area) is a prerequisite for installing these measures..

4) Factors concerning other utilisation of the site or aesthetics

   When planning on-site infiltration measures, it should be considered whether the local original utilisation or function is affected by the installation of the stormwater infiltration measures, such as the basement, foundation of buildings, railways or highways, etc.. The main indicating factor to evaluate these effects is the local groundwater table (described already in the first class) . According to ATV (Commission of Wastewater Technology, Germany), the critical distance between the bottom of the infiltration structure and the groundwater surface is 1 m. But according to other experts this is not necessary, because a sufficient cleaning process occures within the vegetated upper layer

   The effects on aesthetics depend on mainly whether there is enough green space available. Ideally, the design of the infiltration swale or swale-trench system should be integrated into local landscape planning to create pleasant scenery. However, this factor is hardly indicatable.

5) Social and economic factors

   Social acceptance and costs are the final important factors to consider in selecting on-site infiltration measures.

In the above factors, the factors in the first four classes concern spatially distributed information on the planning area and are available in most of cases. They can also be evaluated quantifiably or decisively. Factors in last classes are relevant only by concrete design and construction of the measures. In the planning stage, only factors in the first three classes are relevant.

5.2 Decisions on on-site stormwater management measures for the city of Chemnitz, Germany

5.2.1 Background and goal

The project in this case study is a research project financed by the DBU (German Federal Foundation for the Environment). It is undertaken by following research partners:

- Institut für Wasserwirtschaft, Hydrologie und landwirtschaftlichen Wasserbau, Universität Hannover (IWW)
- Institut für Landschaftspflege und Naturschutz, Universität Hannover (ILN)
- Lehrstuhl für Mikroökonomik, Universität Hannover (VWL)
- Sächsische Landesanstalt für Landwirtschaft, Leipzig (LfL)

The project's main purpose is to develop a new preventive flood control concept for an entire river catchment, which is different from traditional measures such as river engineering construction (excavated river channels, widened river flood plains) and centralised rainwater retention construction (retention basins) etc.. The main approach to the new flood control measures is to reduce runoff and prolong runoff concentration by introducing on-site or decentralised measures (rainwater on-site infiltration in urban areas, conservational cultivation in agricultural areas, etc.) under the consideration of environmental effects and economic cost. The main tasks of this project are:

- First, to analyse and determine the potential for on-site water retention and the preparation of a digital catchment-scale distribution map of water retention potential.
- Second, to select on-site structural stormwater management measures for urban and agricultural areas under the consideration of hydrological, environmental and economic factors, and the preparation of a digital catchment-scale distribution map of water management measures.
- Based on these two maps, scenarios reflecting the innovative flood control concept are formulated for the case study river basin "Mulde Einzugsgebiete" and evaluated through long-time continuous simulation of rainfall runoff process with a hydrological rainfall-runoff model.
- To transform the planning procedures from a local scale to a regional scale

Obviously, the two most important steps of the research can be facilitated by using the developed integrated tools "GMExt + Flext (+ GIS)" of this PhD study. Knowledge bases for evaluating the spatial distribution of water retention potentials and optimal decentralised measures to realise water retention during rain events for urban and agricultural areas, forests and areas with special environmental interests can be formulated by relevant domain researchers. At the same time, the corresponding database containing the data relevant to each evaluation should also be prepared. Along with it, the GIS extension programme GMExt can be used to do automatic spatial intersection among features which contain data of relevant factors separately. Based on the formulated knowledge bases and the prepared databases, automatic evaluations for the whole planning area are carried out with the help of the Flext inference function.

In the following demonstrative application, only the determination of on-site stormwater infiltration measures and preparation of the digital distribution map of measures for the city of Chemitz in Sachsen, Germany is presented.

5.2.2 Stormwater management in Chemniz

Current situation (Umweltbericht 2000 Chemnitz)

Chemnitz is the largest city in Mulde river basin. Rainwater and domestic wastewater in Chemnitz is drained mainly through combined sewer pipes. Separate sewer systems are only found in some recently constructed areas. As most of the sewer pipes are 80-100 years old, the sewer system is relatively overloaded and needs urgent rehabilitation. There is one large wastewater treatment plant responsible for a population of about 320,000 people and several other small treatment plants responsible for around 8000 people. There is a large amount of combined sewer overflow during heavy rain events through about 108 overflow outlets. However, there are currently only 3 stormwater retention basins in operation, so most of the overflow is directly discharged into natural waterbodies. Consequently, the Chemnitz River and other waterbodies are heavily polluted. To meet the German standard of the allowed yearly discharge of COD into natural waterbodies, stormwater retention basins with at least 43.170 m^3 storage capacity should be available. Considering this situation, the responsible authorities formulated on one hand a general plan for the rehabilitation of the sewer system (including enlargement of hydraulic overload sewer pipes, construction of rainwater retention basins, etc.) and on the other hand investigated other alternative ways to solve the stormwater management problem.

A private water management consulting firm was contracted to carried out an investigation on the possibility of installing on-site stormwater management measures in the entire urban area of Chemnitz. Nearly at same time, two pilot projects of installation of on-site stormwater management measures are being carried out (See 5.2.7).

Based on the data collected in the above consulting project, along with a new DEM, an updated decision tree for the selection of on-site stormwater infiltration measures for each local area in the whole urban area of Chemnitz was drawn under the analysis of influencing factors. In contrast to the procedure used in the consulting project, a knowledge base representing the decision tree was formulated in this study. Consequently, an expert system dedicated to the automatic selection of on-site stormwater infiltration measures was developed and integrated into a GIS platform (Geomedia Professional 5.2).

5.2.3 Data bases and processing

Expected heavily polluted areas

The heavily polluted areas in the city of Chemnitz are documented as distributed points in a GIS map. However, there is no information about the size of each area and the type of the pollution. Rainwater infiltration is principally not allowed in areas with questionably large amounts of pollution. Therefore, a spatial distribution layer of the heavily polluted areas was created for further evaluation by transforming the point distribution to aerial distribution. The transformation was to create a circular area around each point in the point distribution layer with a radius of 100 m.

Table 9 Classification of surface slope (Chemnitz)

Slope Class	Slope range	Decentralised rainwater infiltration	Remark
1	0 – 5 %	All measures possible	
2	5 –10 %	All measures possible	infiltration swale should be arranged parallel to surface slope. The construction cost is high.
3	10 –15 %	All measures possible	infiltration swale should be arranged parallel to surface slope. The construction cost is very high.
4	>15 %	rainwater infiltration measure not possible	

Surface slope

Based on the DEM model (20m x 20m) provided by Sächsische Landesanstalt für Landwirtschaft, Lepizig, a sub-DEM model covering the city of Chemnitz was extracted, and a surface slope model for the area of Chemnitz was derived with the help of the spatial analysis function of ARCVIEW 3.1 (GIS environment). The grid slope model was further converted into vector distribution. The real value of the surface slope was classified into four groups that correspond to the constraints of different infiltration measures (Tab. 9).

Table 10 Classification of permeability (Chemnitz)

soil permeability classes	soil permeability (m/s)	Decentralised rainwater infiltration measures	Remarks
Class 1: very permeable	$K_f \geq 1*10^{-2}$	Surface infiltration	
Class 2: very permeable to permeable	$K_f = 1*10^{-2} - 1*10^{-6}$	Surface infiltration Swale infiltration Swale-trench element	Field investigation on soil permeability
Class 3: permeable	$K_f = 1*10^{-4} - 1*10^{-6}$	Aerial infiltration Swale infiltration Swale-trench element	Field investigation on soil permeability
Class 4: permeable to poor permeability	$K_f = 1*10^{-4} - 1*10^{-8}$	All decentralised infiltration measures possible	Field investigation on soil permeability
Class 5: permeable to very poor permeability	$K_f = 1*10^{-4} - <1*10^{-8}$	All decentralised infiltration measures possible	Field investigation on soil permeability
Klass 6: poor permeability	$K_f = 1*10^{-6} - 1*10^{-8}$	Swale-trench system (with throttled drainage)	
Klass 7: very poor permeability	$K_f \leq 1*10^{-8}$	Swale-trench system (with throttled drainage)	
DB,VB,W	Transportation area, waterbody, etc.	Not relevant aera	

Soil permeability value (kf)

Based on the distributed field infiltration tests and a series of geology maps from the state of Sachsen, Germany (1:25000), the spatial distribution of soil permeability in classes was prepared. The available information does not give out accurate value but a wide range on the on-site soil permeability. The boundaries of the ranges are not consistent with the K_f criteria for different infiltration measures (see tab. 8, 10). More than one infiltration measure for certain local areas in regard to the soil permeability is possible. As shown in table 10, areas with permeability classes 4 and 5 will not be assigned a concrete measure but require a field investigation of the infiltration rate. For areas with permeability classes 2 and 3, it is clear that throttled drainage is not necessary. However, whether an extra storage volume should be created with a swale or swale-trench structure still depends on the field permeability test. For areas with permeability classes 6 and 7, infiltration with throttled drainage is necessary. In this distribution map, transportation surfaces like streets, highways and railways are also indicated by assigning special classes (DB, W, VB, etc.) to the K_f for each of them.

Groundwater depth H (m)

As described in the previous section, the depth of the groundwater table is related to the soil water storage capacity and the risk of possible damage to structures by introducing infiltration measures. For this reason, based on available information, the groundwater depth data is available in four ranges in spatial distribution.

Table 11a Classification of groundwater depth(Chemnitz)

Class 1	0-2.5 m
Class 2	2.5-5.0 m
Class 3	5.0-7.5 m
Class 4	>7.5 m

The maximum average depth of the infiltration structures is 1.2-1.5 m. When considering another 1m depth from the bottom to the groundwater table as prescribed in ATV (German Waste Water Technology Committee), the areas with groundwater depth in classes 2, 3 and 4 are appropriate for installation of an infiltration structure. For infiltration measures in areas with groundwater depth in class 1, it depends on the particular situation to determine which infiltration measures are appropriate to be installed. For example, an infiltration swale can be only 30 cm under the earth surface and hence areas with 130cm groundwater depth could be appropriate according to ATV. So, for the areas in class 1, a more detailed field investigation of groundwater depth is necessary.. Finally, the evaluation of groundwater depth is shown according to the following table.

Table 11b Reclassification of groundwater table (Chemnitz)

Groundwater depth in class	Groundwater depth (m)	Rainwater infiltration measures	Remark
Class 1	0-2.5	All measures possible	Field investigation on groundwater table
Class 2	>2.5	All measures possible	

Drinking water abstraction area

In the protected areas (I, II) where groundwater is abstracted for drinking water, rainwater infiltration measures are not allowed because of the potential pollution risk. Hence, these areas are also designated in digital maps.

Soil thickness

Data of soil thickness is not available in the project.

Natural waterbodies

For the evaluation, the area of natural waterbodies is not of interest and hence ruled out from further evaluation. Data of the distribution of natural waterbodies come from the database "CIR" provided by "Sächsischen Landesamt für Umwelt und Geologie", where the spatial distribution of land uses for the whole River Mulde catchment is available. The distribution of waterbodies is simply extracted from the GIS data source as an independent spatial distribution on urban area of Chemnitz.

All these data on the influencing factors are loaded into a GIS map as independent distribution layers. They are distributed in different styles, i.e. the borders of objects in distribution layers are all different from each other. With the help of GMExt, the spatial intersection of all layers with each other creates a new spatial distribution, in which each object contains consistent values of all influencing factors.

5.2.4 Decision tree for selecting measures

In order to systematically analyse the influencing factors of stormwater infiltration measures, a decision logic, i.e. the work flow for the selection of infiltration measures, is made in the form of a decision tree (att. 1). In each node of this tree, one factor is evaluated. The evaluation, namely comparing the practical value against a set of criteria parameters, will lead to one of the possible results. Each result may be a further evaluation or the final decision. The evaluation process starts from the root node of the tree and, depending on the evaluation result of the factor in the current node, another factor may be further evaluated in one of its child-nodes. In this way, all nodes that form a unique path in the tree will be processed until a decision is reached. Decisions for all objects of the output feature of the intersection of all influencing factors features, i.e. the appropriate on-site stormwater infiltration measures (including no possibility of infiltration), can be made systematically by repeating the above evaluation process.

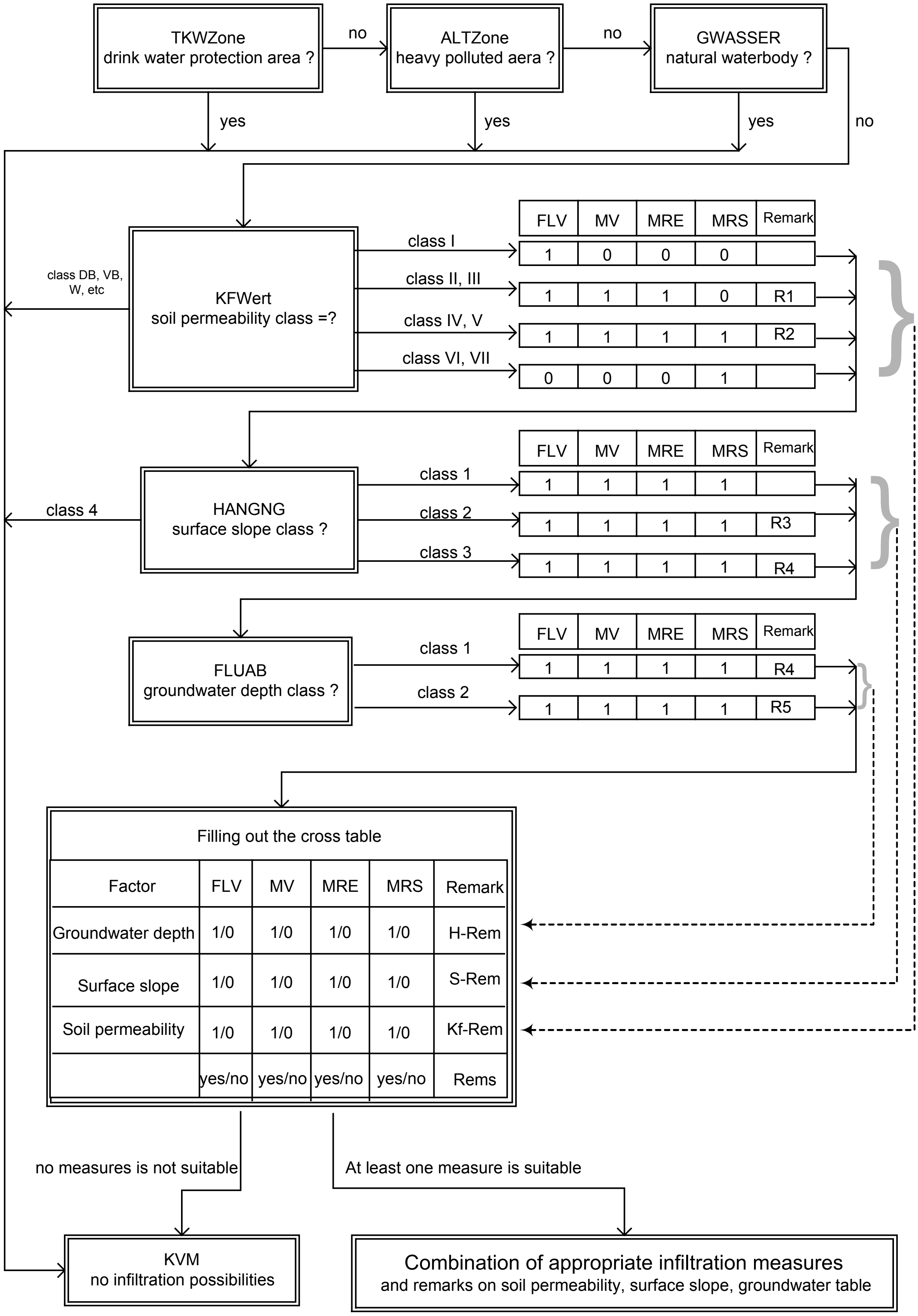


Fig. 23 flowchart of the evaluation on decentral rainwater management (Chemnitz)

In some problems, the final conclusion may have an enormous amount of possibilities (which is referred to as combinatorial explosion in the AI area). If the decision logic is formulated as a strict tree structure, there will be enormous tree branches. In such cases, the decision logic is better formulated as a network of nodes or as a tree in a broad sense. Figure 23 shows the evaluation procedure of the influencing factors in the Chemnitz case, where FLV stands for surface infiltration, MV for swale infiltration, MRE for swale-trench infiltration, MRS for swale-trench-system, and KVM for no infiltration possibility. Each of R1-R5 represents a remark in the corresponding case; each of 1/0 or yes/no or 'Rem' in the cross-table will be replaced with a concrete value during an evaluation of a practical case. The procedure can be classified as follows:

- First, the questions regarding areas protected as drinking water sources, heavily polluted areas and natural waterbodies are asked sequentially. An answer of yes to any of these questions will lead to the conclusion that rainwater infiltration is not suitable.
- Secondly, the soil permeability, surface slope and groundwater depth by class are asked sequentially and, depending on each answer, a table of one row and five columns which corresponds to five different infiltration measures and a remark is filled out with 1/0, where 0 means the relevant measure is denied and 1 means the relevant measure is acceptable regarding the current factor. For soil permeability, an answer of any of classes DB, VB, W, etc. will lead immediately to the conclusion of no infiltration possibility. An answer of class 4 in the question of surface slope will lead to the same conclusion.
- Finally, a cross table is constructed based on the three one-row tables mentioned above. The first row represents the four infiltration measures and a remark respectively. The next three rows come from the one-row tables and the last row represents the conclusion for each infiltration measure. If all fields in a column are filled with 1 (acceptable), then the measure represented in that column is suitable (yes), otherwise it is not suitable regarding the three considered factors. The final conclusion based on this cross table is either no infiltration possibility or the combination of all possible measures and remarks.

5.2.5 Transforming the decision tree into a knowledge base

In order to automate the evaluation process based on the above decision logic, a problem-specific ES should be built with the help of ES tool Flext. Based on the above decision tree, a knowledge base for selecting on-site rainwater infiltration measures in Chemnitz was formulated as follows:

- for each node in the decision tree, creating a corresponding node in the knowledge base,
- for each node in the decision tree in which an influencing factor is evaluated, defining a question asking the data input for that factor, designing a personal interface for the question such that it helps the user to better understand the factor's effects on the infiltration system and to give a better answer to the question. It is important to note that some nodes in the tree can be without questions, like the one "filling out cross table" in this case, which is actually a node of analysis of the results from previous nodes.
- for each path directed away from the current node to another node in the decision tree, formulating a rule (namely, specifying conditions and actions) in mathematical expression.

Node: tkwzone --- evaluation of drinking water source zone

Assign the answer to the question "Is the processed area located in an area protected as a drinking water source?" to the question variable *tkwzone*. The evaluation is then formulated as the following two rules:

- IF *tkwzone* = "yes" THEN no infiltration possibility
- IF *tkwzone* = "no" THEN evaluating for heavy polluted area

Node: altlast----Heavy pollution area

Assign the answer to the question "Is the processed area located in a heavily polluted area?" to the question variable *altzone*. The evaluation is then formulated as the following two rules:

- IF *altzone* = "yes" THEN no infiltration possibility

  Note: in practice, for areas with expected heavy pollution, a field investigation is suggested to check the exact extent of the area and the exact pollution type so that the concrete decisions on relevant infiltration measures can be made. In fact, the "sealed infiltration swale-trench-system" or INNODRAIN (see **5.1.1**) are suitable even if the rainwater comes from heavy polluted area.
- If *altzone* = "no" THEN evaluating for natural waterbody

Natural waterbody

Assign the answer to the question "Is the processed area natural waterbody?" to the question variable *gewa*. The evaluation is then formulated as the following two rules:

- IF *gewa* = "yes" THEN no infiltration possibility. In fact, the natural waterbody is not relevant to the evaluation.
- If *gewa* = "no" THEN evaluating for soil permeability

Soil permeability

Assign the answer to the question “What is the soil permeability class?” to the question variable k_f. The value of *kf* is the permeability class as described before. The evaluation is then formulated as the following rules:

- IF k_f = “I” THEN FLV is OK
- IF k_f = “II” or k_f = “III” THEN FLV, MV, MRE are OK, Remark: field investigation on K_f to finally decide on the optimal measure
- IF k_f = “IV” or k_f = “V” THEN FLV, MV, MRE, MRS are OK, Remark: field investigation on K_f to finally decide on the optimal measure
- IF k_f = “VI” or k_f = “VII” THEN MRS is OK

Note: the measures favoured as a result of this evaluation are not the final decisions. The results must be considered together with the results from the evaluation of surface slope and groundwater depth as described above in the decision tree, namely, filling out a cross table and drawing the conclusion. Therefore, in knowledge base, variables (flv_k_f, mv_k_f, mre_k_f, mrs_k_f, cmm_k_f) are introduced to hold these temporary results, which are represented by 1 or 0 in the fields in the table in the above decision tree. For example, flv_k_f=1 means the measure FLV is OK in regard to soil permeability, and so on. The cmm_k_f holds the remark on soil permeability.

Surface slope

Assign the answer to the question “What is the surface slope class?” to the question variable *hng*. The value of *hng* is the surface slope class as described before. The evaluation is then formulated as the following rules:

- IF *hng* = “1” THEN FLV is OK
- IF *hng* = “2” THEN FLV, MV, MRE, MRS are OK, Remark: infiltration structure must be parallel to surface slope constructed, high cost
- IF *hng* = “3” THEN FLV, MV, MRE, MRS are OK, Remark: infiltration structure must be parallel to surface slope constructed, very high cost
- IF *hng* = “4” THEN no infiltration possibility

Similar to the soil permeability, the results are represented by variables (*flv_hng*, *mv_hng*, *mre_hng*, *mrs_hng*, *cmm_hng*) in the knowledge base.

Groundwater depth

Assign the answer to the question “What is the groundwater depth class?” to the question variable *fluab*. The value of *fluab* is in the groundwater depth class as described before. The evaluation is then formulated as the following two rules:

- IF *fluab* = “1” THEN FLV, MV, MRE, MRS are OK, Remark: field investigation on groundwater depth to finally decide on the possibility of an infiltration measure
- IF *fluab* = “2” THEN FLV, MV, MRE, MRS are OK

Simular to the soil permeability, the results are represented by variables (*flv_fluab*, *mv_fluab*, *mre_fluab*, *mrs_fluab*, *cmm_fluab*) in the knowledge base.

<u>Filling out and analysing the cross-table</u>
The results of the evaluations for soil permeability, surface slope and groundwater depth are represented as variables which make up a cross-table as shown in the decision tree. The cross-table is then analysed through a set of logic mathematic rules. Here, variables *flv*, *mv*, *mre*, *mrs* are introduced to hold the output strings regarding each infiltration measure. The rules are as following:

- IF *flv_kf*=1 AND *flv_hng=1* AND *flv_fluab*=1 THEN FLV is OK (*flv*=“FLV”)
- IF *flv_kf*=0 OR *flv_hng*=0 OR *flv_fluab*=0 THEN FLV is not suitable (*flv*=“”)
- IF *mv_kf*=1 AND *mv_hng*=1 AND *mv_fluab*=1 THEN MV is OK (*mv*=“MV”)
- IF *mv_kf*=0 OR *mv_hng*=0 OR *mv_fluab*=0 THEN MV is not suitable (*mv*=“”)
- IF *mre_kf*=1 AND *mre_hng*=1 AND *mre_fluab*=1 THEN MRE is OK (*mre*=“MRE”)
- IF *mre_kf*=0 OR *mre_hng*=0 OR *mre_fluab*=0 THEN MRE is not suitable (*mre*=“”)
- IF *mrs_kf*=1 AND *mrs_hng*=1 AND *mrs_fluab*=1 THEN MRS is OK (*mrs*=“MRS”)
- IF *mrs_kf*=0 OR *mrs_hng*=0 OR *mrs_fluab*=0 THEN MRS is not suitable (*mrs*=“”)
- IF (*flv*+*mv*+*mre*+*mrs*)= “” THEN no infiltration possiblity
- IF (*flv*+*mv*+*mre*+*mrs*)<> “” THEN output the value of “*flv*+*mv*+*mre*+*mrs*” together with all possible remarks

5.2.6 Results evaluation

Figure 24 shows the results of the execution of the rainwater infiltraion evaluation system. The possible rainwater infiltration measures for different areas in Chemnitz are presented. At meantime, figure 25 shows the remarks from the results of the execution of the expert system. The results can be concluded as following:

- In most of the area in the city, especially in the southeast, MRS, i.e. infiltration swale-trench system, is the only choice for decentralised rainwater management. However, the concrete construction of the structure should be only carried out after determining the local groundwater depth, because in most of this area, the groundwater depth is less than 2.5 m.

- In the River Chemnitz floodplain and around some other natural waterbodies, where the soil permeability is comparatively higher, infiltration swales and swale-trench-elements are suitable for decentralised rainwater management. However, the construction should only occur after further selection between the two after a field investigation of the soil permeability.
- Clearly displayed are areas with no infiltration possibility, which consist of drinking water source zones, heavy industry areas, public transportation areas and areas with high surface slope.
- Surface infiltration is justified practically for no areas.
- In the rest of the city catchment, because of a lack of enough accurate information on the soil permeability, no single infiltration measure is justified. For further planning, the field investigation of soil permeability is necessary.

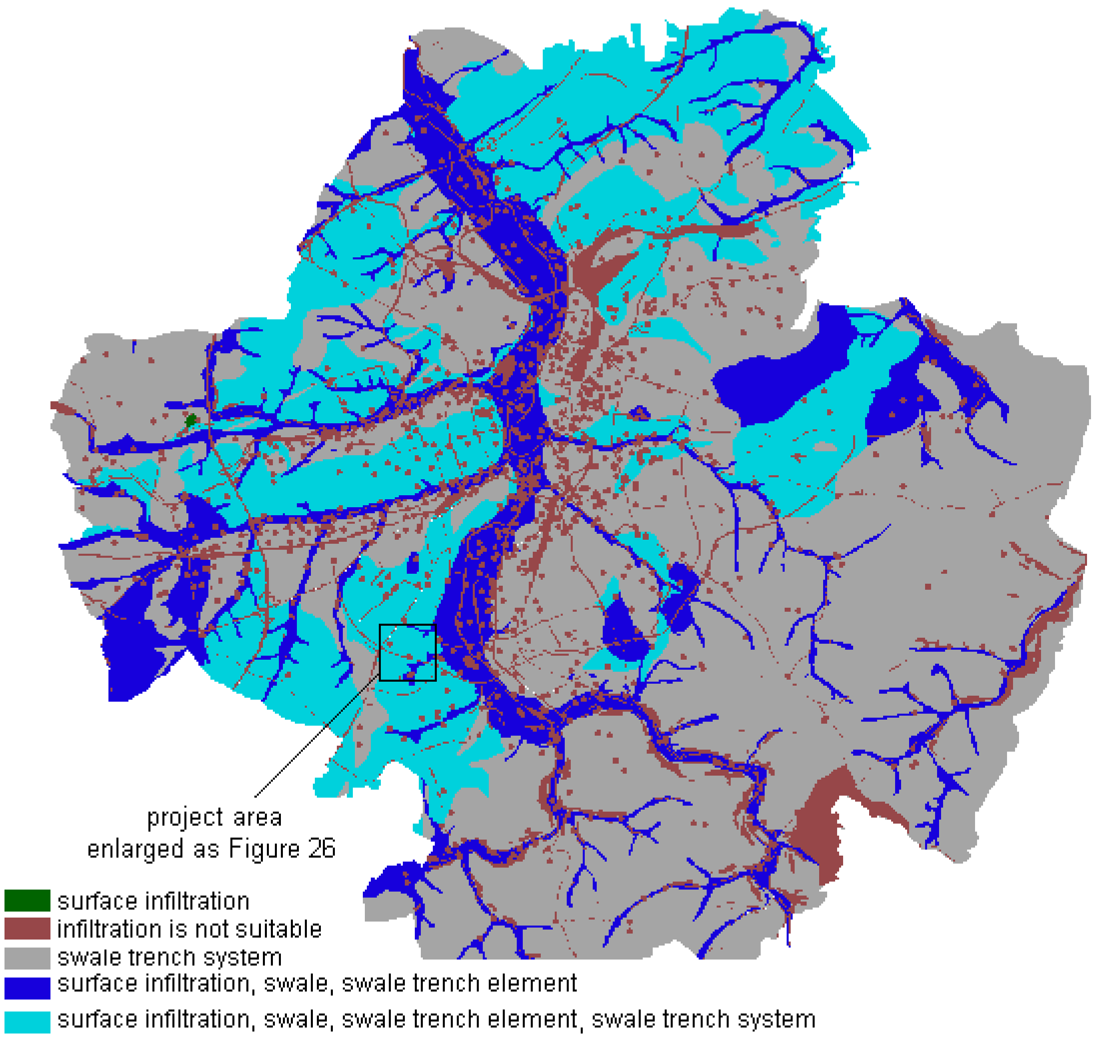


Fig. 24 map of distribution of on-site stormwater management measures (Chmenitz)

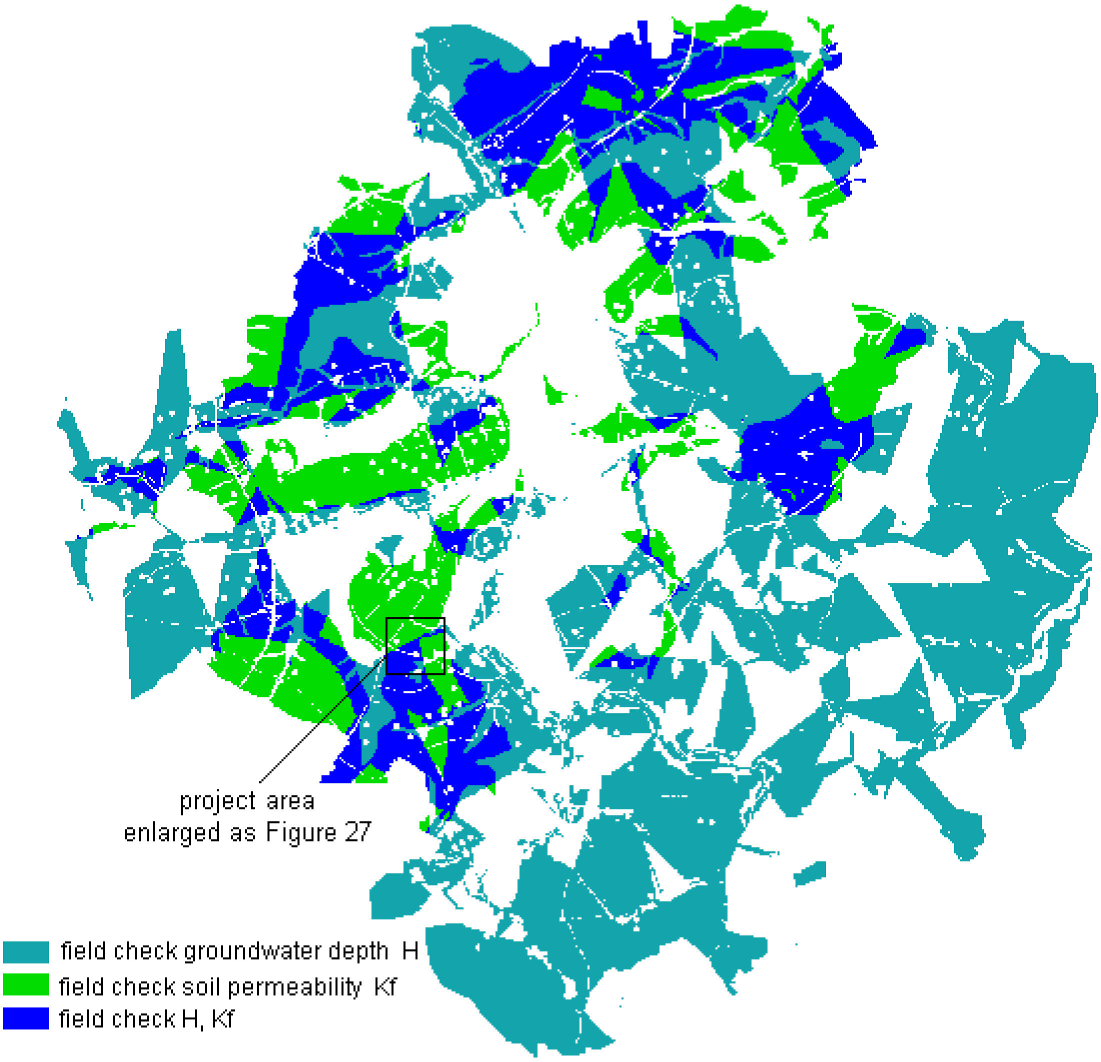


Fig. 25 map of distribution of remarks on influence factors (Chemnitz)

5.2.7 Installation of on-site stormwater management measures in Chemnitz

The concept of on-site stormwater management is no longer just an idea, but a practical alternative method in urban drainage planning of Chemnitz. Because of the advantages, mentioned in 1.1.2, as well as the favourable cost, Chemnitz authorities decided to construct on-site stormwater management structures in a residencial area called Fritz-Heckert (Fig. 26), where rehabilitation was just carried out. This residential area involves two streets with 6 residential blocks. The total area of all blocks is 27,742 m^2 , of which 5,262 m^2 (19%) is roof area which has always been drained by combined sewer system.

To investigate the possibility of on-site stormwater management and to select the best suitable method for the selected project area, an overview map (such as Fig. 26, 27) as a result at the preliminary planning stage is in fact very helpful. These maps indicate not only the possible methods, but also the advice on possible further detailed investigations before design and construction. For example, field measurements or tests both for the groundwater level and for the soil permeability are necessary in this case. In this way, a large amount of work on data collection can be saved because the analyses of factors (such as cost, aesthetic effects, integration into surrounding environment, as well as the data that are necessary even in the preliminary planning stage for analyses for a better overview map, but they simply do not exist, or the accuracy of the data is currently not high enough, etc.) can be postponed until the installation stage, when the focus is actually largely narrowed to the areas where the installation of on-site stormwater infiltration structures has been decided based on the overview map and some other viewpoints. However, this project does not follow this procedure because the area is selected as a sample area for using on-site stormwater management at the same time as the project of investigating on-site stormwater infiltration structures possibilities in the entire area of Chemnitz started. Therefore, the investigation for this project area was independently carried out from scratch and as a result, the swale-trench infiltration elements were selected and built in the residential area (Fig.28).

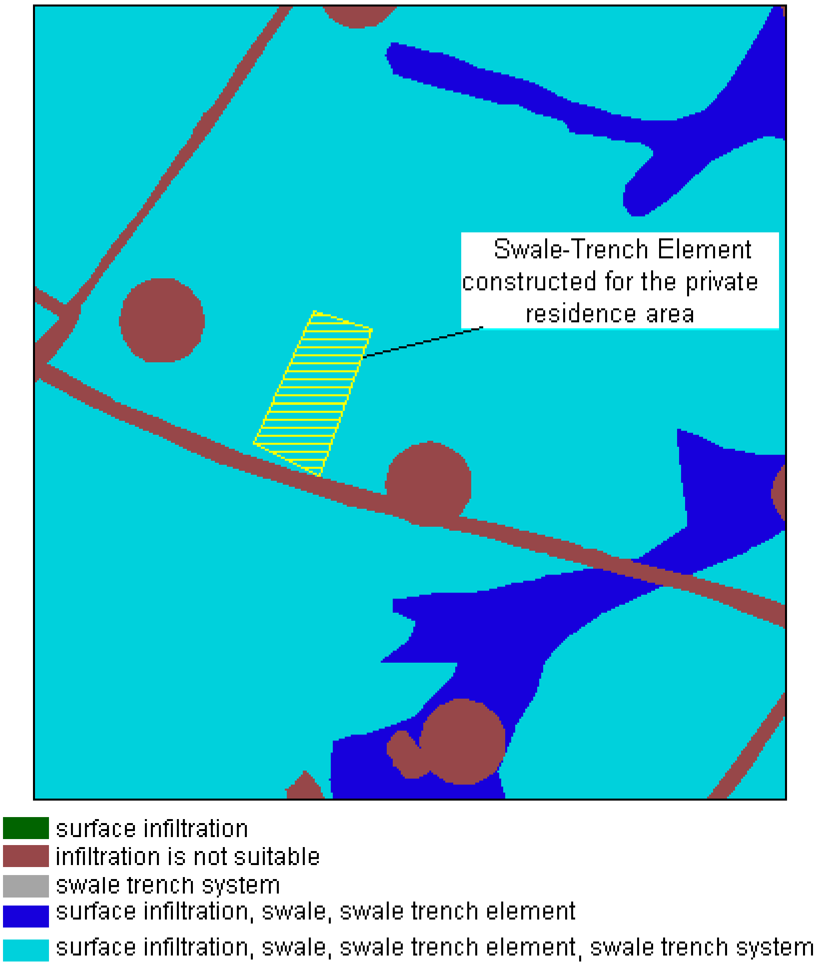


Fig. 26 The zoom view of the project area on the resulting distribution map of on-site stormwater infiltration measures (see fig. 24)

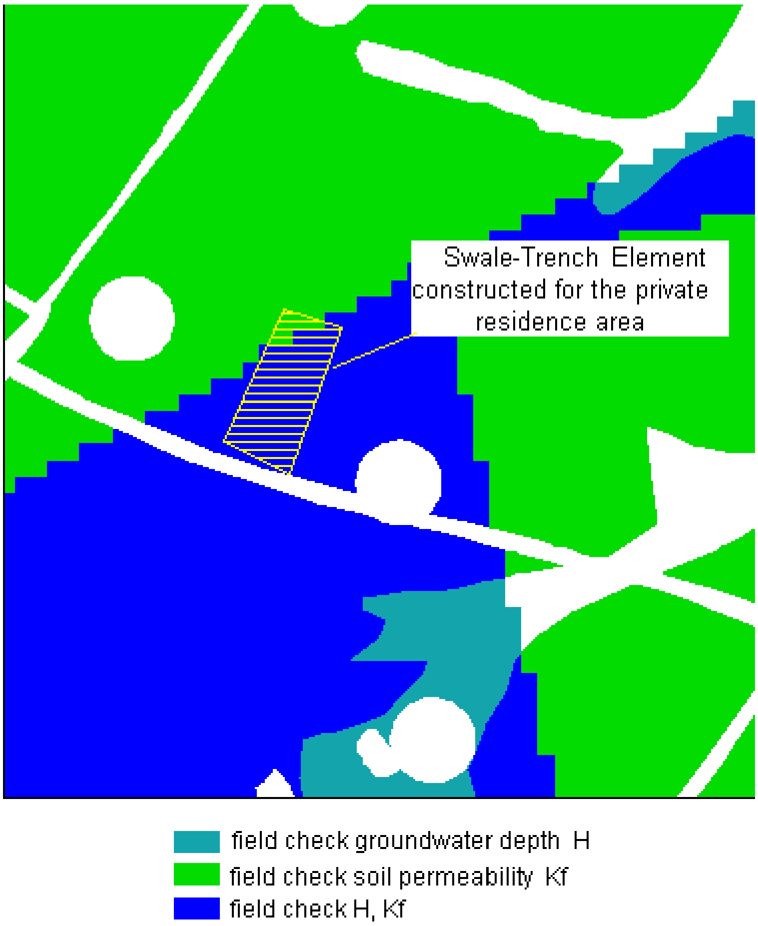


Fig. 27 The zoom view of the project area on the distribution map of rematks on influence factors (see fig. 25)

Other practical constructions of on-site stormwater management measures in Chemnitz are located in the new industrial and commercial area called Röhrsdorf. It involves the stormwater management both for a long-distance transport road and a residential area. Because the Röhrdorf area did not previously belong to the City of Chemnitz, it was not covered by the digital data collected for the investigation and hence is not covered in the resulting map of on-site stormwater management measures (Fig 24). As for the above pilot project, the investigation for this area was also independently carried out from scratch and the INNODRAIN-System was selected for the stormwater management of the transport road.

Fig 28. A Swale-Trench element in residential area Fritz-Heckert, Chemnitz, Germany

Fig 29. INNODRAIN-Elements along a transport road in Röhrsdorf Chemnitz, Germany

5.3 Rainwater management in the Emscher area

5.3.1 Background

The Emscher catchment, with an area of around 865 km^2, is an industial conurbanation in North Rhine-Westphalia, Germany. Due to some historical reasons, such as rapid growth in industry with correspondent large increases in wastewater, land surface subsidence caused by mining, the wastewater is carried via open watercourses, namely, the Emscher and its tributaries, instead of in underground sewer pipes over last 100 years. Now, the Emscher area is no longer being mined, and hence the surface subsidence has come to an end. The Emscher region has decided to convert the open wastewater drainage system to an underground sewer pipe system and the open watercourses to waterbodies of a natural ecological environment. This implies that all wastewater in this region, as well as the stormwater from its intensively sealed urban areas, must be disconnected from the open watercourses. If the new drainage system adopts only the underground sewer system, the disconnection implies further that more than 90% of the rainwater must be totally drained away from the urban area through the sewer system due to the high level of the impervious area in the Emscher region, which has increased by more than 100% over last 50 years. As a result, on one hand the construction of appropriately large sewers, as well as central stormwater structures, such as structures for combined wastewater treatment, flood prevention, etc. is needed. On the other hand, the base flow in the open waterbodies is largely reduced and hence a healthy ecological environment of the water bodies can not be secured. Therefore, for both financial and ecological reasons, the Emschergenossenschaft, the water management authority of the region, has initiated a pilot project “The Boye Ideas Competition”, aiming to investigate other innovative stormwater management ideas. The investigation has been carried out for the Boye catchment of around 80 km^2, which is a tributary of the Emscher. The results of this competition project and numerous other practical investigations has proven that the near-natural stormwater management methods can be realised technically in the Emscher region to partly manage stormwater runoff on-site other than the drainage through the combined sewer system with the substantial advantages both from financial aspects and from the viewpoint of a sustainable open water ecology. The competition results further indicate that the stormwater from 12% to 18% of the sealed area in the whole Boye catchment can be disconnected from sewer system.

To further guide the practice of integrating decentralised near-natural stormwater management methods in the new Emscher wastewater drainage system, a spatial distribution indicating in detail where and which type of decentralised stormwater management method is appropriate is extremely necessary. The following case study contributes directly to this end. The result of this study supports that the methods taken in a current stormwater management project initiated by the Emschergenossenschaft (see 5.37) are appropriate.

As in the Chemnitz case, the problem-specific solution logic and accordingly a knowledge base were built into this demonstration for the evaluation of decentralised rainwater infiltration measures.

5.3.2 Data preparation

Unlike in the Chemnitz case, there were more data available for this investigation, such as soil type, thickness of soil or loose sediment layers, etc. There were also more accurate quantitative values for soil permeability. Therefore in this case, quantities parameters were applied in the relevant rules in the knowledge base. Accordingly there was the opportunity to test the sensitivity of choosing different values as constriant parameter on K_f for the selected infiltration measures. In the following section, the factors introduced in the evaluation process are explained in turn.

<u>Surface slope</u>

The surface slope originates from a DEM model (2 x 2 m, 2001) and is grouped into four classes. The range of the slope in each class is listed with its effect on rainwater infiltration management in the following table.

Table 12 Classification of surface slope (Emscher case)

Class	Surface slope (%)	Effects on decentral rainwater management
1	0 – 5 %	No limitation on infiltration measures
2	5– 10 %	Structures constructed along the surface slope, high cost
3	10 – 15 %	Structures constructed along the surface slope, very high cost, check for economical feasibility
4	> 15 %	Decentral infiltration measures are not economically favorable

<u>Thickness of the soil and sediment layer</u>

Aside from some mountainous areas, the Earth's surface geologically usually consists of loose sediment layers, such as clay, loam, fine sands, corse sands, layers with these mixed materials etc. The existence and thickness of certain layers depends on the geological history.

For decentralised rainwater infiltration, such loose sediment layers with a certain thickness is essential to guarantee that the infiltrated rainwater is further percolated away, although infiltration directly into rocks is theoretically possible. However, the permeability of hard rocks is much more difficult to determine than for loose sediment and the permeability in hard rocks is rather unstable from site to site. In current practice, the thickness of loose sediment is related to the so-called ‘Steinart’ and each type of ‘Steinart’ is estimated as a range of thickness of loose sediment. Their effects on decentralised rainwater management are listed in the following table.

Table 13 Classification of the soil thickness (Emscher case)

Steinart	Thickness of loose sediment layers	Effects on decentralised rainwater management
L, L über L	> 1,5 m	All infiltration measures OK
L über F	0,5 - 1,5 m	Only aerial infiltration and infiltration swale possible
F	< 0,5 m	Infiltration measures not possible

Soil permeability

In this case, the soil permeability is available directly in values. The classification regarding the infiltration measures is similar to table 8.

.

Table 14 Classification of permeability (Emscher case)

Soil permeability		Decentralised rainwater infiltration measure
In m/s	In mm/h	
$\geq 2*10^{-5}$	≥ 72	All measures possible
$5*10^{-6}$ -- $2*10^{-5}$	18 – 72	Infiltration with storage structures (like swale, trench, etc.)
$1*10^{-6}$ -- $5*10^{-6}$	3.6 – 18	Infiltration with underground storage structures (like trench, etc.)
$< 1*10^{-6}$	< 3.6	Infiltration swale-trench-system with throttled connection to sewer system INNODRAIN

Soil type

In order to consider the influence of the perched groundwater regime on rainwater infiltration, the soil type is introduced as an index. In the Emscher catchment, the dominant soil type is clay, which is strongly related to the perched groundwater regime. However, human activities have strongly affected the original situation, i.e. the perched groundwater depth varies greatly from site to site.

Table 15 Classification soil type (Emscher case)

Soil type	Class	Effects on decentralised rainwater management
anthropogenic affected soil	Terrestrial sediment	This type of soil reflects no perched groundwater, so rainwater infiltration is not constrained.
Pseudogley	Terrestrial soil with perched groundwater	In such an environment with high soil water content, complete infiltration is not justified. However, infiltration measures with throttled drainage into natural surface waterbodies are suitable.
	Semiterrestrial soil	In this type of soil, the groundwater depth is between 4 dm und 8 dm. Infiltration measures are hence not suitable. However, the rainwater could be diverted with an open system to suitable sites and infiltrated through decentralised infiltration structures.

Groundwater depth

The groundwater depth is derived from the difference between the surface DEM model and groundwater table model. It then is grouped into three classes: low, middle and high. The range of the depth of each class is listed with the effect on rainwater management as follows.

Table 16 Classification Groundwater depth

Class	Groundwater depth	Effects on decentral rainwater management
Low	> 3m	No limitation on infiltration measures
Middle	1,5--3m	Simple infiltration like a swale is suitable. However, for infiltration measures with underground storage structures, the throttled drainage is necessary.
high	< 1,5m	On-site infiltration is not effective, except MRS

Groundwater protection zone

There is no groundwater protection area in the Emscher catchment.

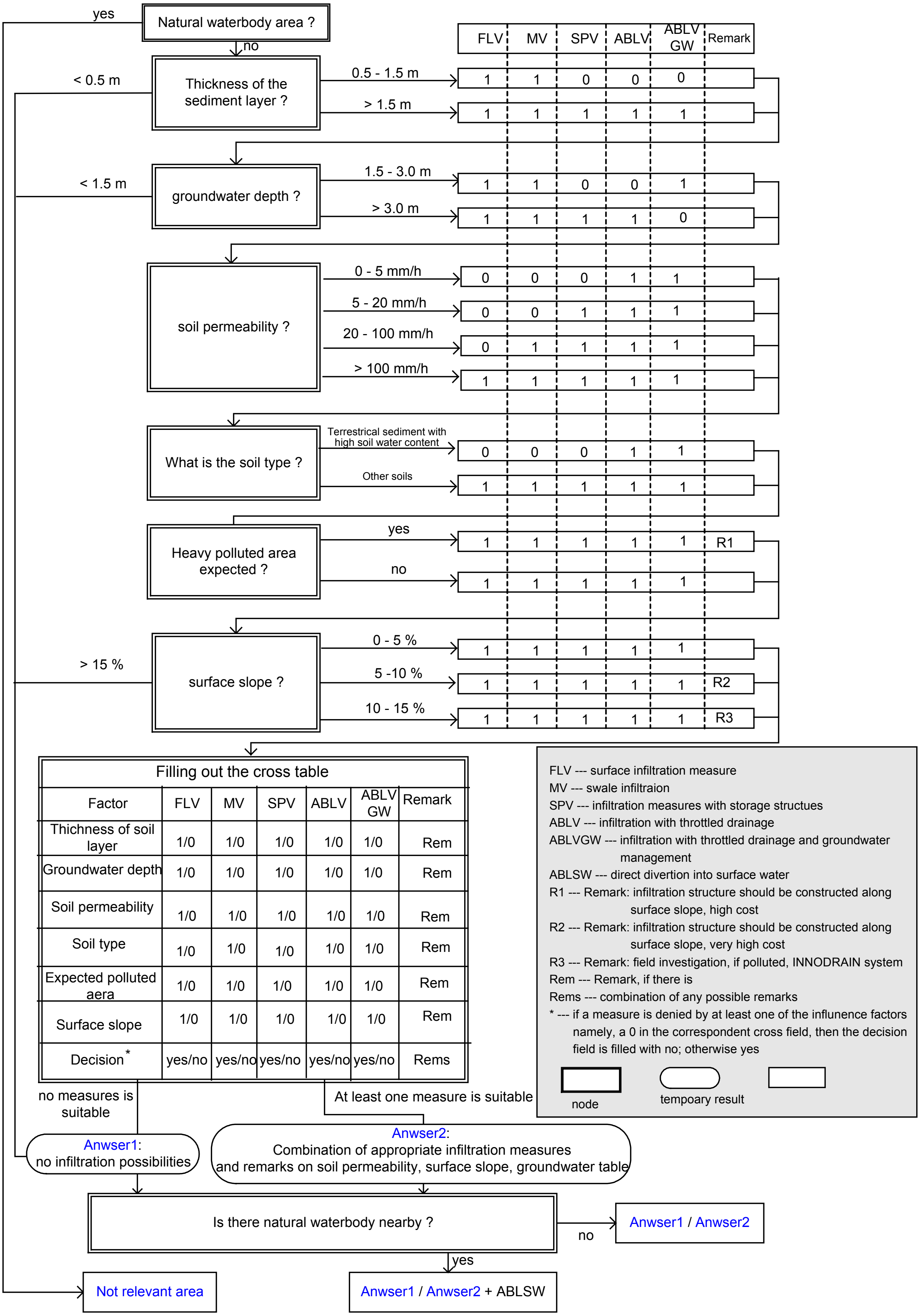


Fig. 30 Flow chart of evaluation on decentral rainwater management in Emescher catchment

Industrial areas

The Emscher catchment is characterised as highly industrialised. Hence, for the investigation of rainwater infiltration measures, the expected heavily polluted areas are prepared in a digital map and are overlapped with other distribution maps of other influencing factors.

5.3.3 Decision tree construction

Based on the classification of the available databases, the decision tree for the evaluation is built as shown in figure 30. In addition to rainwater infiltration measures, the direct drainage of rainwater into nearby natural waterbodies is also considered in this evaluation.

5.3.4 Knowledge base

Based on the decision tree, the knowledge base is formulated in the same way as in the Chemnitz case.

5.3.5 Evaluation of results

The results of the evaluation automated by the expert system tool Flext with the above-mentioned knowledge base are presented in figure 31, in which the suitable rainwater infiltration measures with their related remarks are spatially displayed in different colours and musters. The explanation for each colour and muster is given in detail as follows.

All on-site infiltration measures possible

Due to the lack of adequate accurate data on soil permeability, it is necessary to investigate on-site soil properties in detail to better select a definite infiltration measure.

Infiltration with water storage structures

Due to the local soil permeability, only the decentralised infiltration measures with storage structures are suitable.

Infiltration with water storage structures, occasionally groundwater manahement

Due to the local soil permeability, the decentralised infiltration measures must have storage structures, for example, swale infiltration. If the underground storage structures, like trenches, should be constructed due to the limited available area on the surface, groundwater management is necessary.

Swale infiltration

Due to the local soil permeability, the decentralised infiltration measures must have storage structures. However, the thickness of the soil or loose sediment layers is too small here, so only the swale infiltration measure is possible.

Infiltration measures with high storage capacity

Due to the low soil permeability, The infiltration measures must have a high storage capacity; swale infiltration is possible only if a large natural area is available.

Infiltration structures with high storage capacity and throttled drainage

Due to the low soil permeability, complete infiltration is here not possible, infiltration measures must have both storage structures and throttled drainage.

Infiltration measures with high storage capacity and throttled drainage in combination with groundwater management

Due to the low soil permeability, complete infiltration is not possible here. The infiltration measures must have both high storage capacity and throttled drainage. In addition, groundwater management is necessary.

On-site rainwater infiltration is not suitable

Due to various reasons:

- High groundwater table
- Soil layers too thin
- Soil permeability too low

on-site infiltration is not suitable.

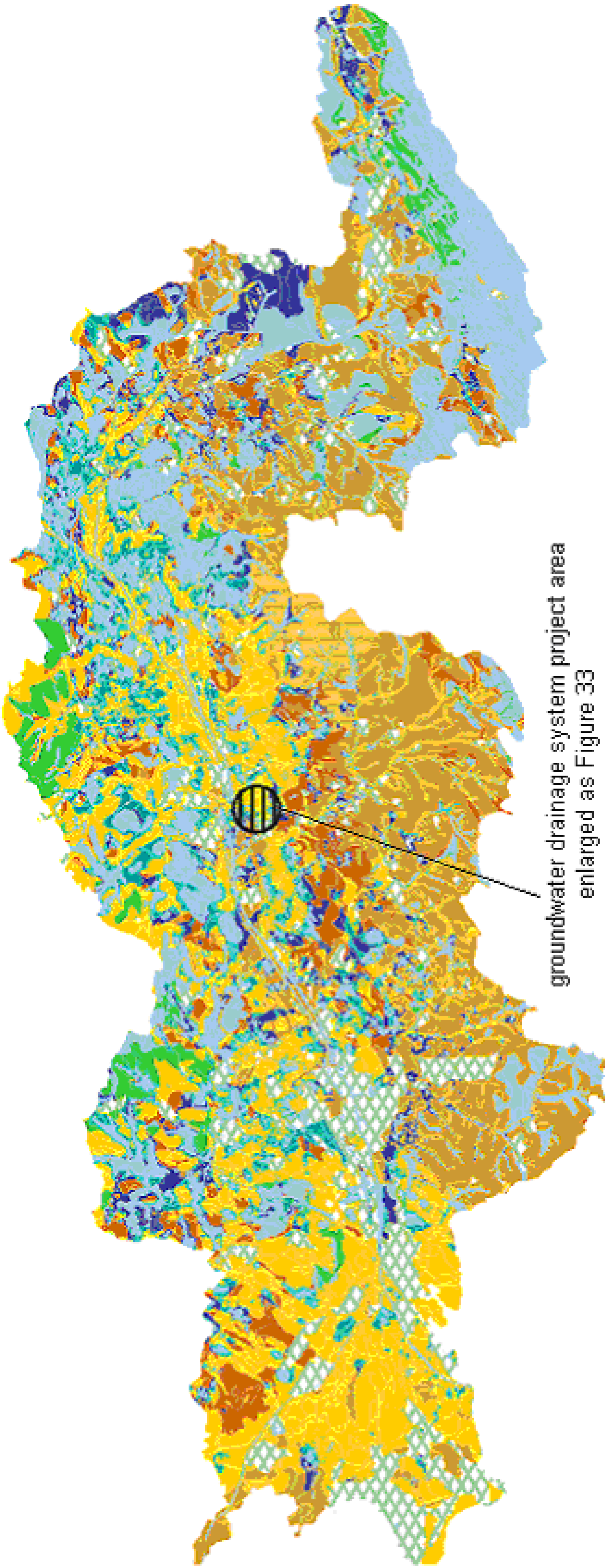


Fig. 31 Themeatic presentation of on-site stormwater infiltration measures for Emscher catchment

5.3.6 Sensitivity analysis

The soil permeability is the most important influencing factor for selecting rainwater infiltration measures. Hence, the parameters applied in the constraints for the evaluation of the effects of the soil permeability on infiltration measures are critical to the final results. For a given geological catchment, the analysis of the sensitivity of the results of the evaluation of rainwater infiltration measures to the applied criteria parameters will certainly be informative to planners. In Table 14, there are three parameters in the constraints which sort the decentralised rainwater infiltration measures into four classes, namely, measures with necessary combination with throttled drainage, measures with at least both surface and underground storage structures (swale, trench), measures with at least surface storage (swale) and measures with no limitation. Clearly, the most significant parameter is the lowest one which indicates whether the measures should be combined with drainage, or in other word, whether the investigated area can possibly be managed with complete infiltration. Figure 32 shows the change of area in which throttled drainage is necessary for any potential infiltration measures with the lowest criteria parameter. It is shown that the evaluation results are sensitive when this parameter is larger than 12 mm/h. Therefore, the selection of a parameter of 3.6 mm/h (which is based on some simple hydrological calculations and hence relatively empirical) is actually conservative because using a slightly smaller or larger parameter does not influence the results.

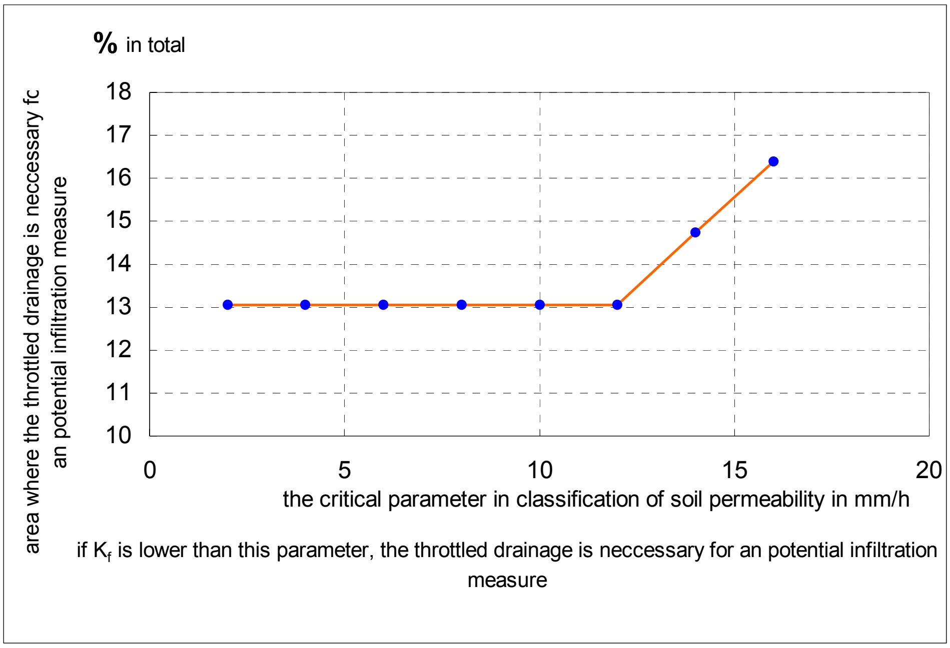


Figure 32 The sensibility of decentralised rainwater infiltration measures on the critical parameter in soil permeability classification

5.3.7 Realisation of on-site stormwater management methods in combination with groundwater managemnet in Herne

In polder areas of the Emscher catchment, due to the unsealed sewers and drains in private buildings, no seepage damage to basements, etc. has previously occurred. However, to prevent the sewer system from the high external seepage, sewers in this region will be rehabilitated so that they are water-tight and private groundwater drains will no longer be allowed to be connected to the sewer system. Therefore, another groundwater drainage system must be developed to control the potential seepage problem and even surface flooding from groundwater flow. For the construction of such a system, direct use or integration of the existing open water ponds and trenches is a favourable choice.

Now that a groundwater drainage system is to be constructed, it can further serve the stormwater mangement in that the throttled overflow from decentralised infiltration structures (for example, infiltration swales, trenches) is drained through this system. This combination makes for the best use of the developed drainage system. The integration of stormwater management, on one hand, has reduced the combined sewer flow, and on the other hand has increased the base flow of the open waterbodies integrated in the general drainage system. Another positive effect is that the infiltrated stormwater will dilute the local groundwater with high sulphate content and PAK (in some zones).

The Emschergenossenschaft planned to construct this combination system of stormwater and groundwater management, called an infiltration-drainage system, in a sub-catchment Herne (Fig. 31, 33) as a pilot project. It is important for the authority to realise its general goal of disconnecting stormwater from 15 percent of the total sealed area in the Emscher region from the sewer system because, on one hand, the combination of groundwater management must be considered in a decentralised stormwater management plan in many sub-catchments in the Emscher region where the seepage problem is serious. On other hand, it will make the best demonstration of the combination of decentralised stormwater management with other measures.

Although the decision for the combination of the stormwater and groundwater management in this pilot project is not a direct consequence of the results of this study, that is, the distribution map indicating the possible on-site stormwater management measures in the entire Emscher cathcment (Fig. 31), it is comfirmed by the map. On the contrary, this planning confirms the accuracy and demonstrates the usefulness of the map. An enlargement of the map for this project area is shown in Fig. 33. In this figure, the study suggests that the stormwater infiltration measures must be combined with storage structures. If a surface storage structure is limited by the available natural area and underground storage structures (trenches) must be constructed, then groundwater management should be considered in some areas. Groundwater management is not recommended for most of the project area in the map because the evaluation is based on the current situation, namely on current data on groundwater levels, which would be higher after the rehabilitation of the sewer system for watertight sewer pipes and the disconnection of private groundwater drains from the sewer system.

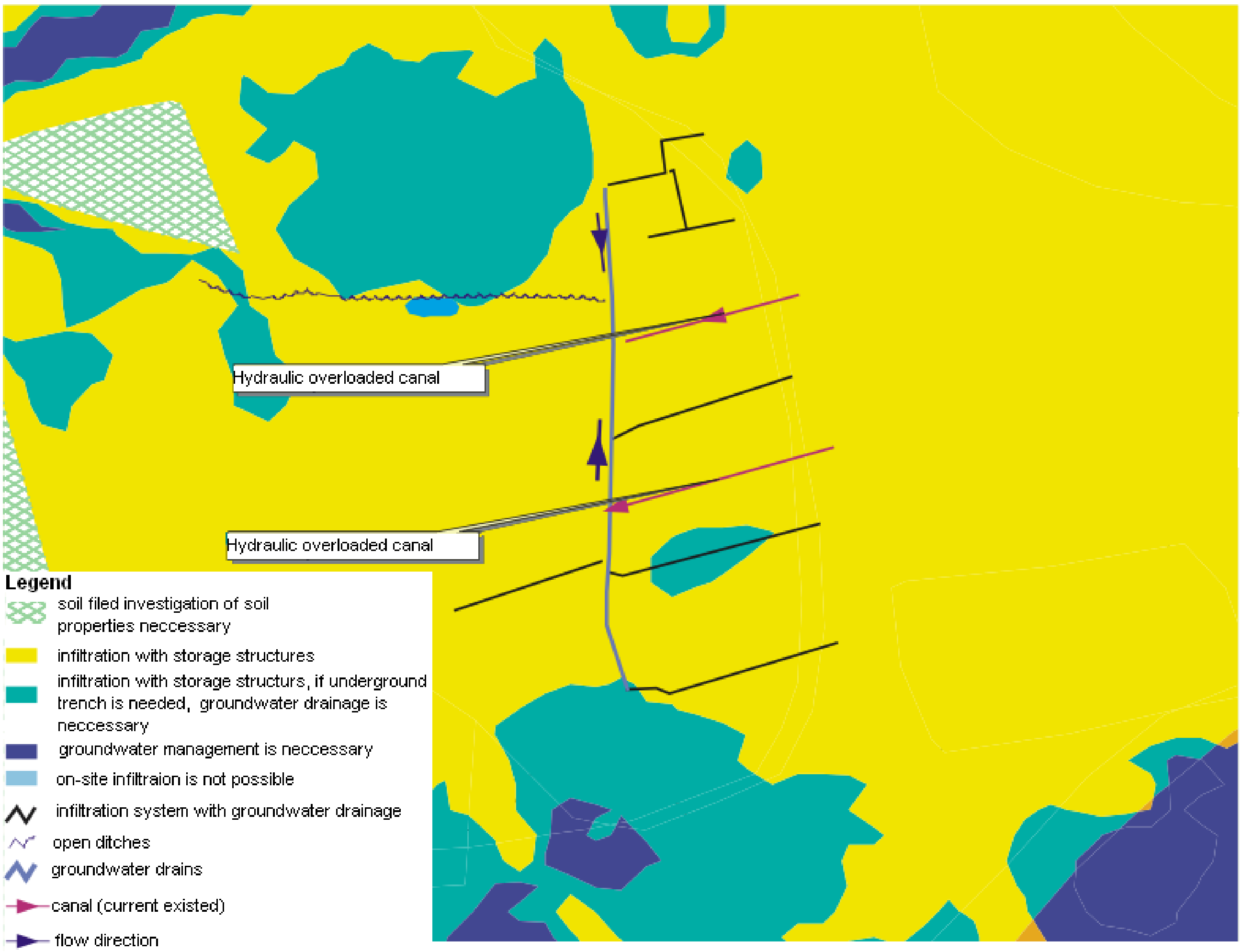


Fig. 33 Enlarged map of on-site stormwater management measures for the Herne area and schematic presentation of the planned infiltration-drainage system (source: Emschergenossenschaft, IPS Germany, unpublished)

6. Discussion and prospects

6.1 Further application

The developed two problem-specific expert systems can be applied as a decision support system in other on-site stormwater infiltration planning projects. It may be necessary to slightly modify the pre-defined knowledge to meet the project's specific demands, such as, due to lack of data in certain factors, introducing new factors or simply changing the different critical values of some constraints on certain measures. However, the possibility of modifying the pre-defined knowledge is characteristic of this transparent and open system. In fact, this model is to be applied by respective research fellows to building expert systems or decision support systems for the analyses of decentralised water management measures on agricultural land and measures with considerations for environmental protection in the previously mentioned preventive flood control project. In general, the integrated model (Flext + GIS) can be used for developing GIS-based rules systems for entirely different problem domains.

A simple, important application may be the development of an intelligent end of a decision support system in which the complicated decision-making procedure can be formulated as decision trees or networks of nodes and the different professional models or functions necessary in relevant steps can be activated in the corresponding rules of a formulated knowledge base. The advantages of using Flext in this case are:

- Complicated decision-making procedures can be clearly formulated as structured trees or networks.
- For the dialog in each decision-making procedure, an informative graphic interface can be designed.
- Easy integration into a GIS platform, if necessary.

6.2 Limitations and further extension

The present version of Flext lacks the possibility of dealing with uncertainties, which is in fact unavoidable in practice when only an estimation for a certain constraint factor, instead of an accurate measurement, is available. For example, in the developed system for the evaluation of on-site stormwater infiltration, the answer to "Is the area to be connected heavy polluted?" may be "maybe". In this case, a factor indicating the probability of a clearer or more accurate answer (such as "yes" with a probability of 0.6 and “no” with 0.4) can be introduced and the inference engine should be extended to consider this factor in its condition-checking mechanism. In fact, in a full fuzzy system, not only the data or facts, but also rules may be attached with probability factors. The probability factors to each input data may be explicitly specified by users during the execution of a developed system at the human-computer dialog. It can also be calculated by the system itself according to a pre-defined model for a collection of pre-defined vague vocabulary (such as “maybe”, ”likely”, ”hardly”, ”probably”, etc.). This calculation is usually called defuzzlation. Aside from the ability to process uncertainty, it may also be necessary to add some self-learning ability to the FLEXT knowledge formulator, although it is now already very simple and easily understandable. However, an overall learning ability is still a great challenge to AI reaserch. Current existing learning systems are mostly concentrated in a narrow, strictly-defined way.

7. User manual on expert system tool Flext

7.1 Knowledge base formulator

7.1.1 Variables

Variables are the basic elements holding information in the Flext knowledge base. They are used only in ***global scope*** and have only three types: ***Number***, ***String*** and ***Boolean***. The type is declared simply by writing the variables delimited by comma in respective text fields in the variable declaration interface of the knowledge base formulator (att. 2). The definition can also be divided into sentences for clarity. The name of a variable can be any combination of Roman letters and Arabic numerals.

7.1.2 Functions and expressions

The operators " +, - , *, /, ** " are applied for numerical operation.

The operator " + " is also applied for string combination operation.

The operator " = " is applied for setting values of variables of all types.

The operators " >, >=, <, <=, =, <> " are applied in comparison of two variables of numerical value.

The operators " =, <> " are also applied in comparison of two variables of string type.

The logic operators "&& (means AND), || (means OR)" are applied in Boolean expressions.

Functions available for numerical computation are currently: **int, float, str, val, abs, sin, cos, sqr, exp, ln, log, not, isnull, eof**. Following are exemplary explainations of these functions.

abs(-3.6)=3.6

int(3.6)=4

int(-3.6)=-4

str(3.6)="3.6"

val("3.6")=3.6

sin(3.14159/6)=0.5

cos(3.14159/6)=0.866

sqr(9)=3

exp(1)=2.71828

log(10)=1

ln(10)=2.30

2**3=8

isnull(*x*), see if variable *x* is NULL or not, return TRUE or FALSE, value can be put into a boolean variable which can be used further in conditions of rules

not(*x*), negative of boolean variable *x*, for example:

x=(9>8), *x* has value TRUE

y=not(*x*), *y* has value FALSE

x,y are boolean type variables.

eof("*db1*"), see if an file pointer is at the end of the opened file, return TRUE or FALSE, value can be put into a boolean variable which can be used further in conditions of rules; *db1* is a handle to a opened database file or text file; see **OpenDatabas**e, **OpenTextFil**e. Note that the handle should be closed with "".

In addition, functions are written in mathematical convention, but with parentheses for each operation with priority. For example, *x=a+(b*c)*, *x=c + cos(a)*, *y=(b>c) || (c <=0)*. If the parentheses are omitted, then the operation will be sequentially computed without consideration of conventional priority. For example, x=a+b*c is not processed as x=a+(b*c), but actually processed as x=(a+b)*c. For more than two string variable computation, it is not necessary to put parentheses around each of the two arguments because the operation priority here is not relevant. For example:

prints=(question+answer+remarks)

7.1.3 Nodes

In Flext, a knowledge base consists of nodes, which are further constructed from rules and questions. The main interface of the knowledge base formulator is table-edit mode (Fig.12) and graphic-edit mode(Fig.13).

Table edit mode

In the table-edit mode, the whole screen is similar to a table for definition of the knowledge in one node. The node name must be specified in the node name text field. A node name can be any combination of Roman letters and Arabic numerals. Under the node name text field is the main frame, where rules and questions of the node can be specified. The rule fields are divided into two parallel columns: condition fields and action fields. There are five pairs of condition and action fields. However, by using the UP and DOWN buttons to the right of action field, as many rules as necessary can be edited in one node.

By clicking on a condition field or action field, another frame (att. 5) is displayed with two rich text edit fields for conditions and actions respectively. There are also two list boxes with items for operators and keywords used in conditions and actions respectively. Conditions must be specified in one sentence, while the actions may have more than one sentence; each begins with a keyword indicating a specific action.

Three question fields are displayed on screen. However, as for the definition of rules, as many questions as necessary can be edited by using the buttons UP and DOWN to the right of the questions layout button. Each question contains a field for specifying the name of a variable that is designated to hold the answer of the question during runtime. The type of question variable must be assigned before it is referenced here. For each question, one can specify the allowed answers.

For string type variables, the allowed answers should be written and delimited by a comma. Leaving the boundary field blank or writing NULL NULL means any answer inputted at runtime is allowed.

For number type variables, the lower boundary and upper boundary should be specified and delimited by a space. Writing NULL in the lower or upper boundary means the corresponding boundary is unlimited.

For Boolean type variables, the boundary input is not relevant because the allowed answer is fixed, namely, true or false.

For each question, a text sentence should be formulated in the question text field. The **Background** button activates the interface for design of a question-specific graphic interface (see **att. 6**). However, the design of a question interface is optional.

At the top of this table-edit interface, buttons for navigating from node to node (Previous, Next), adding a new node (Insert Before, Insert After), and deleting a node (Delete) are available.

Graphical edit mode

In the graphical edit mode, the knowledge base is displayed as a decision tree or a network of nodes. A node in the graphical network corresponds to a node in the knowledge base. Nodes are linked by the **GoTo** action in their relevant rules. Actions which do not contain a **GoTo** sentence to direct the current node to another node will also be displayed as a special node (end node) of the nodes network. The name of the end node is actually the action of the relevant rule.

When the **add node** option on the toolbar frame is checked, one can click the mouse on any point of the network to add a new node. This consequently initiates the input frame for entering the name of the new node. If the new node is designed to be an end node, i.e. as the action of a new rule, then the action sentence is written in the input text field. In the same way, but with the **edit node** option checked, one can edit the node name or the action sentences of a rule, i.e. the name of an end node.

The linear connection between two nodes with an arrow indicates there is/are a rule/rules between the two nodes. By checking the **Edit Rule** option and then clicking the mouse on the start node and the target node (i.e. the node to which the arrow points) sequentially, the condition and action of a rule between these two nodes can be edited in a popup frame, which is the same as the one accessed by clicking the condition or action field in table edit mode. In writing the action sentences, the **GoTo** sentence, which directs the current node to another node, does not need to be added because the direction is already implied by clicking on the target node. Similarly, one can add a new rule between two nodes by checking the **Add Rule** option.

By checking the **Add Question** or **Edit Question** option, one can click on any node other than an end node to add or edit a question in the node. The question input frame is similar to the one in the table edit interface.

When checking the **DelNode** option, one can click on a node to delete it and all related rules.

When checking the **DelRule** option, one can delete a rule by clicking on the start node and target node sequentially.

When the check box **Full Text** is checked, conditions and actions of all rules are displayed on the nodes network.

When the **Move** option on the toolbar frame is selected, the position of a node can be arbitrarily moved by clicking on the node, holding the mouse and dragging. In the same way, but by clicking the mouse on any point other than a node, one can move the whole node network on the screen. The nodes can be zoomed in windows by right-clicking the mouse to select the first corner of the zoom window, holding and dragging the mouse towards the second corner point of the zoom window. One can also zoom the nodes by checking the **Zoom** option on the toolbar frame and clicking on the screen. In this case, the zoom rate under one click can be exactly given.

When the knowledge base is formulated as a tree structure, this tree can be structurally displayed in a tree control both in table and graphic edit modes. From this tree control, one can have an overview of the knowledge base structure and can easily access any node in the current knowledge base.

7.1.4 Rules

Conditions

An condition contains a complex Boolean expression which may contain mathematic operations and functions. For example,

((x>y) && (a=(b+cos(c+6)))) || (anwser= “yes”)

Actions

Actions of a rule can contain many sentences; each sentence represents an action indicated by a keyword; the syntaxes related to all keywords are listed in section **7.16**. Actions are executed sequentially. However, any sentence after **GoTo** or **Conclusion** will be neglected because these two actions either leave the current node or end the reasoning process.

Note: in the formulation of rules, the following case should be avoided.

Condition : (x>5)

Action: Set x=(x+9)

Such a rule will lead to an endless loop.

Rules like this:

Condition : (x>5)

Action: **Set** x=(x+9)

GoTo node2

will work because it will leave the present node immidieately after the addition.

7.1.5 Question interface

In this frame, the question-specific background to be displayed to the user at runtime can be designed. Here, formatted texts, simple geometry shapes (line, rectangle, circle, ellipse), can be added or edited. Graphics from files can also be inserted. However, in the knowledge base, only the file names of the graphics are saved, so the referenced graphic files must also be copied to the same directory where the knowledge base is. The general background colour of the question dialog interface can also be assigned.

Similarly, one can design a face page for his problem-specific expert system. The menu command **Face Page** accesses the same design interface as the one used for question background design (att. 6).

7.1.6 Keywords

Set: assigns a value to a variable. The value can be a constant or the value of another variable or function. In one **Set** sentence, more than one value assignment action can be specified, but they must be delimited with a comma.

For example: **Set** x=88, name=”John”, a=b, c=(x+a)

GoTo: changes the node with the rule with the GoTo action from active status to inactive; changes the target node from inactive to active. However, other currently active nodes remain active.

For example: **GoTo** node2

Suppose that this is an action of one rule in *node1* and that *node1* and *init* (another root node, see 4.3.4.2) are currently active. If the action is executed, then *node1* will become inactive and *node2* will become active. The node *init* remains active.

Conclusion: stops the system and outputs the results. The results can be constants or values of variables, which should be delimited with a comma. In the dialog execution of the system, each output constant or variable value is printed in one row. In the execution based on the database, the outputs of values of variables are written in different fields in the record set of the database. The field should be matched with the variables to be outputted through the menu command **Database Input** of the knowledge base formulator. The variables that are present in a **Conclusion** sentence but not matched to fields in the record set of the database selected for necessary inputs will be neglected in output. The following is an example of a **Conclusion** sentence.

Conclusion "This is an example output", "x=",x, "y=",y

Open: Opens a standard document (such as a Microsoft Word document, a website, a graphic file, etc.) independently with a default executable programme. For example:

Open c:\untitled.doc

Run: executes an external programme and suspends the inference process until the end of the external program. For example:

Run c:\programme\notepad.exe

Run c:\programme\notepad.exe untitled.txt

OpenDatabase / CloseDatabase: opens / closes a database's record set, for example:

OpenDatabase c:\test.mdb table1 **AS** db1 (Access Database)

Or

OpenDatabase c:\test.dbf **AS** db1 (Dbase format)

CloseDatabase db1

ReadRecord / EditRecord: reads / writes the current record of an opened record set. For example:

ReadRecord db1 x=Filed1, y=Filed2, z=Filed3...

EditRecord db1 Filed1=x, Filed2=y, Filed3=z...

MoveRecord: moves the record pointer to the previous / next / first / last record. For example:

MoveRecord db1 **PREVIOUS** / **NEXT** / **FIRST** / **LAST**

DeleteRecord: deletes the current record of an open recordset.

DeleteRecord db1

AddRecord: adds a new record to an open recordset, for eaxmple:

AddRecord db1 Filed1=x, Filed2=y, Filed3=z...

SearchDatabase: searches a database's record set for the first record in which the specified conditions are satisfied.

SearchDatabase db1 criteria

Criteria is SQL-like query string, such as "field1=*x* and field2=*str1* or field3=999"

Where *str1* is a string variable, *x* is a numeric variable

If a search is successful, then the first matched record becomes the current record; otherwise, the pointer is at end of the record set.

OpenTextFile: opens a text file for input and output. For example:

OpenTextFile test.txt **FOR READ** / **WRITE** / **APPEND AS** txt1

ReadTextFile / WriteTextFile: reads from / writes to a text file. For example:

ReadTextFile txt1 x1,x2,x3

Txt1 is the handle to an opened text file. X1,x2,x3 are system variables.

ReadTextFile reads once a row from the text file. The data in a row should be separated by commas; strings should be closed by “”

WriteTextFile txt1 x1,x2,x3

The same as **ReadTextFile**, but writes to a text file

CloseTextFile : closes an open text file. For example:

CloseTextFile txt1

7.1.7 Setting input from a database

As mentioned before, the formulated knowledge base can be repeatedly executed automatically. The necessary (question) inputs of each execution will automatically be read from different fields of a record in a database recordset. The results of each execution will also automatically be written to the designated fields of the corresponding record. However, the questions and the outputted variables must be related to the designated fields of a database recordset in the formulation of knowledge base.

Through the menu command **Database Input**, a frame for matching questions and output variables can be accessed. (att. 4) . There are two pairs of list controls in the frame. The upper pair of list controls displays the questions and the output variables in the knowledge base and the fields in the selected database recordset respectively, while the lower pair of list controls displays the matched questions and output variables with their corresponding fields.

7.2 Execution of an expert system

When a knowledge base is created or loaded in the knowledge base formulation platform, it is ready to be executed either in interactive mode or in the database.

7.2.1 Interactively running an ES

The popup menu command **Interactive** starts the inference module of Flext which prepares all variables, nodes (rules, questions) and stand-by functions in the special storage structure (see **4.3.4.2**) and displays the designed face page. Further confirmation by clicking the command button **Start** in the lower right corner of the screen will start a concrete reasoning process. In this interactive mode, the questions in the knowledge base are asked in a sequence designated by the reasoning structure of the knowledge base. Next to a question, the allowed answers for that question are listed. The user needs only to click on the appropriate item to answer the question. If the allowed answers for a question are not specified, the user needs to type the answer in the input field. During the system execution, the current reasoning steps being executed can be viewed by clicking the menu command **Tracing**, which suspends the execution and accesses the graphical edit interface to display the knowledge base and trace the path through the nodes completed so far. By clicking the pop-up menu **Interactive**, one returns to the inference module and continues answering questions until a conclusion is reached. The outputted contents (constants, values of variables) are printed in a rich text field. By clicking the command button **Return** in the lower right corner of the screen, the start page of the current expert system is displayed again.

7.2.2 Running an ES for GIS

The popup menu command **Run for Gis** will start the inference module to execute the expert system repeatedly based on the data from a database record set. However, before the inference module is started, it is asked which database is to be used for inputs. If the database is an MDB database, further selection of a table is needed. After the database (and table, for MDB database) is selected, the inference module prepares the knowledge base in a special storage structure and displays a progress bar and a text field which shows the progress of the processing, i.e. the currently processed record. One execution processes one record. The execution can be started from a desired record of the database record set by specifying it in the text field and clicking the command button **Start** under the progress bar. The execution can be stopped at any time by clicking the command button **Break** (att. 11)

Literature

Albertsen, M. (1993). ein Expertensystem für Fluid-Rock-Interaction-Probleme in Wasser- und Erdölbohrungen, DGMK-Bericht, 436-1, Deutsche Wissenschaftliche Gesellschaft für Erdöl, Erdgas und Kohle

Abu Noman M. Ahsanuzzaman *et al* (2004). Simple Expert System for Evaluation of Nutrient Pollution Potential in Groundwater from Manure Application, International Journal of Geomechanics, Vol. 4, No. 4, December 2004, pp. 285-295

Bolder, J. / Lindiger, T. / Zech, H. (1998). Planung und Bau des Retentions-Bodenfilters 'Rieselfeld' in Freiburg, Zeitschriftenaufsatz: Korrespondenz Abwasser (1998)

Bruce G. Buchanan, (2002). Brief History of Artificial Intelligence: MODERN HISTORY. At American Association for Artificial Intelligence Website: http://www.aaai.org/AITopics/bbhist.html#mod, visited at 16 Oct. 2004

Beecham, S.; Kandasamy, J.; Khiadani, M.; Trinh, D. (2005): Modelling On-Site Detention on a Catchment-Wide Basis, Urban Water Journal, Pages: 23 – 32, Publisher: Taylor & Francis

Ben Urbonas, P.E. (1997). Design & Selection Guidance for Structural BMPs , Sustaining Urban Water Resources in the 21st Century , pp. 242-262

Booty, W.G., D.C.L. Lam, A.G. Bobba, I. Wong and G.S. Bowen (1990). An Expert System for Water Quality Modelling. In: Proceedings, Canadian Hydrology Symposium on Expert Systems and Statistical Methods in Water Resources, CCIW, Burlington, Ontario, Canada, pp. 63-66.

Cawsey, A. (1998): The Essence of Artificial Intelligence.

Chang T. J. et al (1996). Development of an Expert System for Daily Drought Monitoring , Journal of Computing in Civil Engineering, Vol. 10, No. 1, January 1996, pp. 20-24

Chang, T. J. ; Moore D. (1994). Reservoir Operation by the Use of an Expert System, Proceedings of the 1994 Conference: Hydraulic Engineering, pp 97-104

Chang, J., Moore, D. (1997). An Expert System Approach for Water Management in Case of Drought, Konferenz-Einzelbericht: IIS, Intelligent Information Systems, 1997

Chau, W. ; Chen, W. (2001). An Example of Expert System on Numerical Modelling System in Coastal Processes , Zeitschriftenaufsatz: Advances in Engineering Software

Chin K. K., Ng K. Y., *et al* (1998).Urban Renewal and Stormwater Management Planning in an Urban Center, Proceedings of the 25th Annual Conference on Water Resources Planning and Management held in Chicago, IL, June 7-10, 1998, pp. 22-28

Comas, J et al (2002). The STREAMES Project: Linking Heuristic and Empirical Knowledge into an Expert System (ES) to Assess Stream Managers, Proceedings of Conference: iEMSs 2002 Integrated Assessment and Decision Support, June 2002, Lugano, Switzerland, p 444

Crum, J. M. ; Mulvihill M. E. (1996). A Knowledge Based System for the Design of Open Channels, Proceedings of North American Water and Environment Congress & Destructive Water, pp. 4125-4130

Culbert, C., Riley, G., Donell B. (Jun. 1995), CLIPS Reference Manual: Volume I The Basic Programming Guide, Volume II The Advanced Programming Guide, Volume III The Interfaces Guide

David C. Froehlich (1996). Searching for Optimal Combinations of Stormwater Detention Basins , North American Water and Environment Congress & Destructive Water 1996, pp. 2264-2269

Daene C. McKinney, *et al* (1993). Expert Geographic Information System for Texas Water Planning , Journal of Water Resources Planning and Management, Vol. 119, No. 2, March/April 1993, pp. 170-183

Daniels, K. (1994). Integrale oder ganzheitliche Planung - Ein Muß für die Zukunft? Zeitschriftenaufsatz: Elektrowärme International, Edition A (Elektrowärme und Technischer Ausbau)

Gimbel, R. et al (2002). Modellierung mehrstufiger Trinkwasseraufbereitungsanlagen mittels eines expertensystem-basierten Simulationsmodells (METREX) am

Beispiel von Oberflächenwasser, Abschlussbericht/durchführende Institutionen Gerhard-Mercator-Universität Duisburg, Fakultät für Ingenieurwissenschaften, Institut für Informationstechnik

Doneker *et al* (1991). Expert Systems for Mixing-Zone Analysis and Design of Pollutant Discharges, Journal of Water Resources Planning and Management, Vol. 117, No. 6, November/December 1991, pp. 679-697

Feigenbaum E. A. (1977). The Art of The Artificial Intelligence: Themes and Case Studies of Knowledge Engineering. In Proc. 5th International Joint Conference on Artficial Intelligence, pp. 1014-29

Fuchs L., Neumann S. (1988). Ein lernendes Expertensystem zur Steuerung städtischer Kanalnetze, Z. f. Stadtentwässerung und Gewässerschutz, Heft 2, p 1-19

Glenn A. Bottomley (1995). Using SWMM in Urban Stormwater Master Planning , Integrated Water Resources Planning for the 21st Century , pp. 13-16

Gómez M., Sánchez H., Vázquez S. (2002). A Simple Expert System for Initial Size Estimation of Detention Basin, in Proceedings of the Ninth International Conference on Urban Drainage

Hahn, M. A. et al (2002). Expert System for Prioritizing the Inspection of Sewers: Knowledge Base Formulation and Evaluation, Journal of Water Resources Planning and Management, Vol. 128, No. 2, March/April 2002, pp. 121-129

James A. Bachhuber (1996). A Decision Making Approach for Stormwater Management Measures—A Case Example in the City of Waukesha, Wisconsin, North American Water and Environment Congress & Destructive Water, pp. 3981-3986

James Crum and Michael Mulvihill (1991). An Expert System for the Planning and Design of Flood Control Channels, Proceedings of the 18th Annual Conference and Symposium: Water Resources Planning and Management and Urban Water Resources, pp. 429-433

John E. Hopcroft; Jeffrey D. Ullman (Addison-Wesley, 1979), Introduction to Automata Theory, Languages, and Computation

Jackson P. (1999), Introduction to Expert Systems, Addison Wesley Longman Limited 1999

Larry A. Roesner, P.E., et al (1997). Integrating Stormwater Management into Urban Planning in Orlando, Florida , Sustaining Urban Water Resources in the 21st Century , pp. 509-535

Lam, D.C.L., D.A. Swayne, I. Wong and J. Storey (1990). A Knowledge-Based Approach To Regional Acidification Modelling. In: Proceedings, Canadian Hydrology Symposium on Expert Systems and Statistical Methods in Water Resources, CCIW, Burlington, Ontario, Canada, pp. 12-15.

Lam, D.C.L. and C. Pupp (1996). integrating GIS, Expert System, and modelling for State-of-Environment Reporting, pp. 419-422. In: M.F. Goodchild et al. (ed.), GIS and Environmental Modelling: Progress and Research Issues. GIS World Books, Fort Collins, Col.

H. K. Lee, et al (1997). Fuzzy Expert System to Determine Stream Water Quality Classification from Ecological Information, Asia Waterqual' 97, ISSN 0273-1223, p. 199-206

León C. et al (2000). EXPLORE–Hybrid Expert System for Water Networks Management, Journal of Water Resources Planning and Management, Vol. 126, No. 2, March/April 2000, pp. 65-74

Leibundgut, ch.; Eisele M. *et al* (2005). Catchment Oriented Assessment of Runoff and Solute Dynamics - Development of The Assessment Procedure 'Hydrological Quality' into an Expert System for River Basin Management, at Project Website:

Http://Www.Hydrology.Uni-Freiburg.De/Forsch/Hydgue/Hydgue-E.Htm, Visited on Feb. 2005

Liong S. Y. et al (1991). Knowledge-Based System for SWMM Runoff Component Calibration, Journal of Water Resources Planning and Management, Vol. 117, No. 5, September/October 1991, pp. 507-524

Lundgren, P. S. ; Barber, M. E. (1993), Development of an Expert System for Urban Runoff, Proceedings of the Symposium: Engineering Hydrology, pp. 1084-1089

Marti A. Hearst and Haym Hirsh, (2000) AI's Greatest Trends and Controversies, at Website: http://www.computer.org/intelligent/articles/AI_controversies.htm, Visited at Apr. 8, 2005

Marissa Koop (2001), "Urbanisation and Urban Development in China", at website: http://www.accci.com.au/koop.htm, visited Aug. 18, 2004

Merta M., Seidler C., Uhlenbrook S., Tilch N., Zillgens B., Kirnbauer R. (2003): Das wissensbasierte System FLAB als Instrument zur prozessbezogenen Raumgliederung in mesoskaligen GebietenTagungsband zum Tag der Hydrologie, 20.-21. März 2003; Kleeberg (Hsrg.): Klima - Wasser - Flussgebietsmanagement - im Lichte der Flut. Forum für Hydrologie und Wasserbewirtschaftung, Band 1, 171-178.

Nir, D.(1991). Expert Systems for Agricultural Water Systems, in Decision Support Systems: Water Resources Planning, Edited by D. P. Loucks and J. R. da Costa. NATO ASI Series, Springer-Verlag, Berlin, Germany, pp. 461-470, 1991.

Nils Thorsten Lange (2003) Einsatz von Methoden des "Soft Computings" in der Hydrologie, Bericht 15, Fachgebiet Wasserbau und Wasserwirtschaft Universität Kaiserslautern

New Jersey Department of Environmental Protection, "New Jersey Stormwater Best Management Practices Manual", at Website:

http://www.njstormwater.org/tier_A/bmp_manual.htm, Visited on Oct. 15, 2004

On-Site Stormwater Management Manual Auckland City, 1.0 Purpose and Applicability, at Website:

http://www.aucklandcity.govt.nz/council/documents/onsite/default.asp, Visited on Apr. 16, 2005

Papa, F.; Adams B. J.; Guo Y. (1998). A Family of Analytical Probabilistic Models for Urban Stormwater Management Planning, Proceedings of the 1998 National Conference on Environmental Engineering, pp. 404-409

Ram D. Sriram (1997) Intelligent Systems for Engineering: A Knowledge-based Approach, Springer-Verlag London Limited 1997

Reffat, R. M.; Harkness E. L. (2001). Expert System for Environmental Quality Evaluation, Journal of Performance of Constructed Facilities, Vol. 15, No. 3, August 2001, pp. 109-114

Schneider, F. (1999): Messtechnische Untersuchung und modelltechnische Beschreibung des Speicher- und Abflussverhaltens von Mulden-Rigolen-System, Dissertation, Veröffentlichung in Vorbreitung, Institut für Wasserwirtschaft, Universität Hannover

Shepherd, A. ; Ortolano L. (1996). Water-Supply System Operations: Critiquing Expert-System Approach, Journal of Water Resources Planning and Management, Vol. 122, No. 5, September/October 1996, pp. 348-355

Sieker, F. (1999): „Geographical Information systems used as a Planning-Tool for On-Site Stormwater Management Measures", Proceedings of NATO-Advanced research Workshop "Coping with Floods: Lessons Learned from Recent Experience", Czech Republic

Sieker, F. (2000) Investigations of the Effects Of On-Site Stormwater Management Measures In Urban And Agricultural Areas On Flood, Flood Issues in Contemporary Water Management, Kluwer Academic Publishers 2000, pp 303

Sieker F.; Wilcke, D.; Akkermann, M.; *et al* (2002). Innovativer Ansatz eines vorbeugenden Hochwasserschutzes durch dezentrale Maßnahmen im Bereich der Siedlungswasserwirtschaft sowie der Landwirtschaft im Einzugsgebiet der Lausitzer Neiße, DBU Projekt AZ 15877

Sieker, F.; Sieker, H. (Hrsg.) (2002): Naturnahe Regenwasserbewirtschaftung in Siedlungsgebieten – Grundelagen und Anwndungsbeispiele, Neue Entwicklungen – 2. Neu bearbeitete Auflage –Kontakt & Studium Band 508, Renningen-Malmsheim, expert verlag

Sieker, F; *et al* (2004). Regen(ab)wasserbehandlung und -bewirtschaftung unter Berücksichtigung der Anforderungen nach 7a WHG und einer möglichst ortsnahen Versickerung , Umweltforschungsplan des Bundesministeriums für Umwelt, Naturschutz und Reaktorsicherheit

Sieker, F., Zimmerman, U., Sommer, H. Entwicklung eines Planungsinstrumentes zur flächenbezogenen Regenwasserbewirtschaftung unter besonderer

Berücksichtigung des vorbeugenden Hochwasserschutzes.Wasserwirtschaft 95 (2005) Nr. 3: 7-14.

Sieker H., Bandermann S., Zimmerman U. (1999) Geographical Information Systems Used as a Tool for Decentral Stormwater Management. Proc. of 8th Int. Conf. on Urban Storm Drainage, Sydney, Australia, 30 Aug.-3 Sept. 1999

SIEKER, H. (2001); Generelle Planung der Regenwasserbewirtschaftung in Siedlungsgebieten, Mitteilungen des Instituts für Wasserbau und Wasserwirtschaft, TU Darmstadt, Heft 116, 2001

Sieker, H., Förster, M., Scheibel, M., Zimmerman, U. DAYWATER – ein Entscheidungshilfesystem für die Bewirtschaftung von Regenwasser.Wasserwirtschaft 95 (2005) Nr. 3: 15-22.

SIEKER, H. *et al* (2002): Ideenwettbewerb Boye. Alternative Planungsvarianten zum Umbau des Emschersystems am Beispiel der Boye. Dahlwitz-Hoppegarten

Sieker H.; Bandermann S., Deckert A., (Jan. 2005) Potenziale dezentraler Regenwasserbewirtschaftung im Einzugsgebiet der Berne

Simonovic, S. P. (1991). Knowledge-Based Systems and Operational Hydrology, Canadian J. Civ. Eng., 18(1), 1-11, 1991.

Simonovic, S. P. (1993). Flood Control Management by Integrating GIS with Expert Systems, in *Application of Geographic Information Systems in Hydrology and Water Resources Management,* Proc. of the Vienna conf., April 1993, Edited by K. Kovar and H. P. Nachtnebel, IAHS Publ. No. 211, pp. 61-74, 1993.

Sojda, Richard S (2003). Decision Trees within Expert Systems for Modelling Ecological Knowledge Related to Water Level Management for Palustrine Wetlands, 2003 Seatle Annual Meeting, "Wetland Science: Intersection of Hydrogeology, Geomorphology, Ecology, and Computer Modeling", (Nov. 2-5, 2003)

Sommer, H. *et al* (2002):"Messtechnische Untersuchung zur Funktion eines Mulden-Rigolen-Systems", GWF-Wasser/Abwaaser, 143. Jahrgang, Heft 12

STECKER, A. & BANDERMANN, St. (1996): „Auswahl und Klassifizierung relevanter Einflussfaktoren auf die dezentrale Regenwasserbewirtschaftung und

Anwendung des Geographischen Informationssystems IDRISI als Planungshilfe“. Stadtentwässerung und Gewässerschutz (SUG), H.37.Swayne, D.A., D.C.L. Lam, A.S. Fraser and J. Storey (1992). Chemical Environmental Models Using Data of Uncertain Quality. In: *Int. J. Man-Machine Studies*, 36, pp. 327-336.

STECKER, A. [1997]: Anwendung der Mulden-Rigolen Kombination zur dezentralen Regenwasserbewirtschaftung in urbanen Entwässerungssystemen, Schriftenreihe für Stadtentwässerung und Gewässerschutz, Band 16, SUG-Verlag, Hannover.

Steffen P. Meyer et al (1993). Geographic Information Systems in Urban Stormwater Management , *Journal of Water Resources Planning and Management*, Vol. 119, No. 2, March/April 1993, pp. 206-228

Stiles, J. M. ; Punnett, R. E. (1993). Including Expert System Decisions in a Numerical Model of a Multi-Lake System Using STELLA, Proceedings of the 20th Anniversary Conference “Water Management in the ’90s: A Time for Innovation”, pp. 487-490

Streames, European intl. Project (2000-). Website: http://streames.org/index.html, visited at March 2005

Stuart Russell and Peter Norvig. (Addison-Wesley, 2002), Artificial Intelligence: A Modern Approach

Svendsen, M. und Balachandran, C. (1993) A Flexible Expert System Based Typology of Irrigation System Types, proceedings of “Management of Irrigation and Drainage Systems: Integrated Perspectives, Sponsored by the Irrigation and Drainage Division of the American Society of Civil Engineers on July 21-23, 1993 in Park City, Utah”, pp. 850-858

Teegavarapu R. S.V. et al (2003). MIST (Model Identification and Selection Tool): A Knowledge-Based System for Selection of Water Quality Models, Proceedings Of The Fourth Joint International Symposium On Information Technology In Civil Engineering, November 15-16, 2003, Nashville, Tennessee, pp. 1-11

Teutsch G., Barczewski B., Kobus H. (1991). Bewertung von Grundwaserprobenahmetechniken zur Erkundung und Überwachung von Altlasten. Inst Wasserbau Univ, Stuttgart, Konferenz-Einzelbericht: Altlastensanierung '90, Dritter Internationaler KfK/TNO Kongreß

Thomas R. Sear and Ronald L. Wycoff (1993). Selection of Optimal Best Management Practices (BMP's) , Water Management in the '90s: A Time for Innovation, pp. 747-750

Thorsten Jonas (2000). Entwicklung eines Expertensystems zur Auswahl geeigneter Regenwassaerbewirtschaftungskonzepte im urbanen Raum

The Drools Development Team. Introduction: Working With Knowledge. The website of Drools, http://drools.org/ , visited on Dec. 18, 2004.

The American Association for Artificial Intelligence , AI Topics: a Dynamic Library of Introductory Information about Artificial Intelligence, Expert Systems, Representation, Reasoning, http://www.aaai.org/AITopics/html/welcome.html, visited at Aug. 2004

United Nations Environment Programme (2005). "Key Facts About Cities, Issues for the Urban Millennium, United Nations Environment Programme", at website:

http://www.unep.org/wed/2005/english/Information_Material/facts.asp

William L. Magette and Adel Shirmohammadi (1992). Expert System for Agricultural and Water Quality Management, Proceedings of the Irrigation and Drainage Sessions at Water Forum '92, pp. 442-447

William N. Lane (1995). Blue-Green Revisited: Integrating Stormwater Management Into the Urban Planning and Development Process , Integrated Water Resources Planning for the 21st Century , pp. 29-32

Yang, S. T. ; Zhu, Q. (2003). A Model of Intelligent Interpreting Soil Erosion Based on Geographical Knowledge, International Geoscience and Remote Sensing Symposium, IGARSS 2003 Learning From Earth's Shapes And Sizes, p 2468-2470

Yoon, J., Padmanabhan, G. (1995). Heuristic Knowledge-Based Tool for Rainfall Synthesis, Runoff Estimation and Hydrograph Generation. Heuristische wissensbasierte Methode zur Regensynthese, Abschätzung des Ablaufs und Hydrographenerstellung, Zeitschriftenaufsatz: Transactions of the ASAE (1995)

Zhu, X. X.; Simpson, A. R. (1996). Expert System for Water Treatment Plant Operation, Journal of Environmental Engineering, Vol. 122, No. 9, September 1996, pp. 822-829

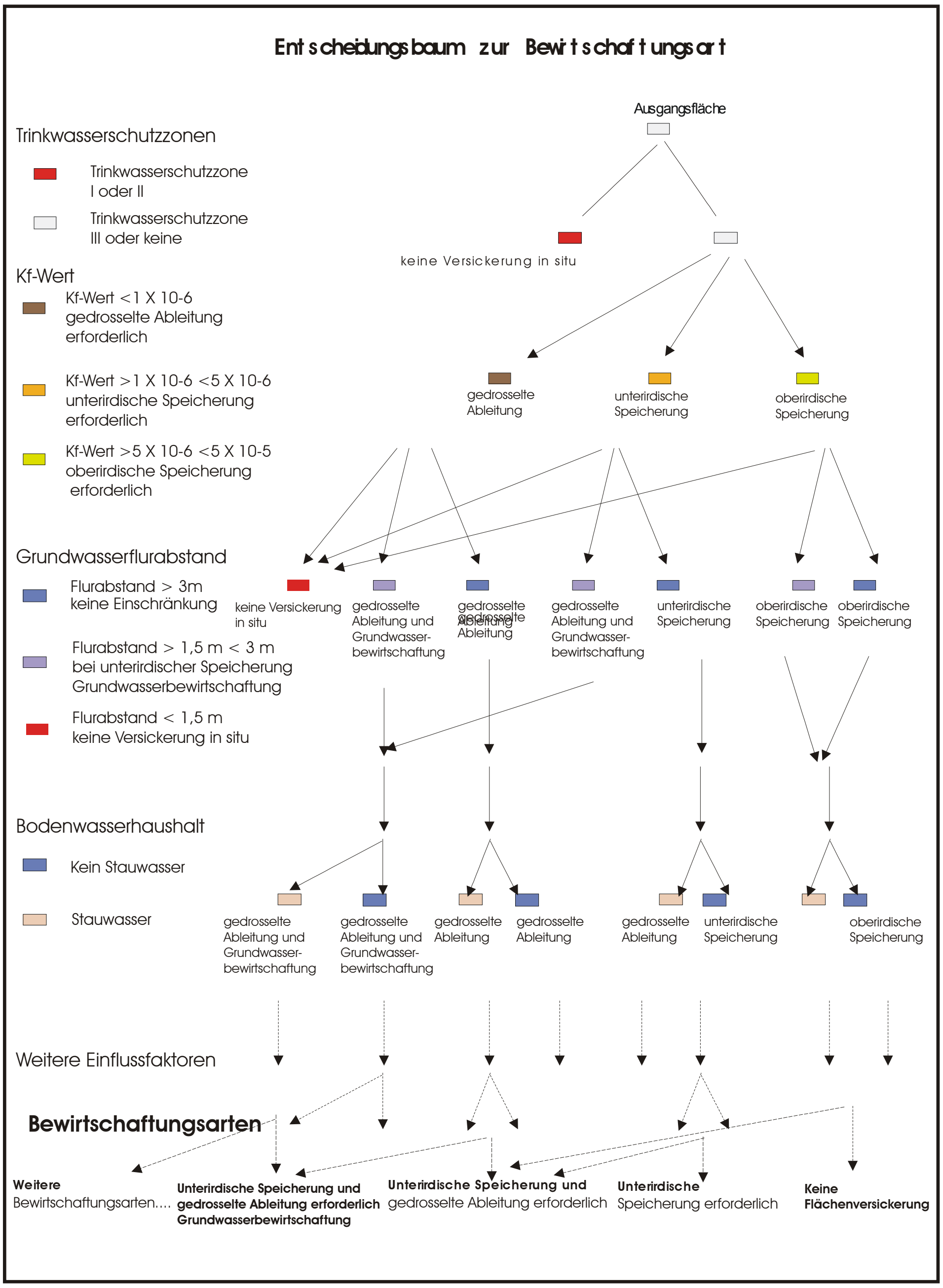
ZHU, X.; HEALEY, R.G.; ASPINALL, R.J. (1998). A Knowledge-Based System Approach to Design of Spatial Decision Support Systems for Environmental Management. Environmental Management 22 (1): 35-48

Zimmermann, S. (1999). Regionalisierung in der Hydrologie, Dissertation at Internationalen Hochschulinstitut Zittau, Germany

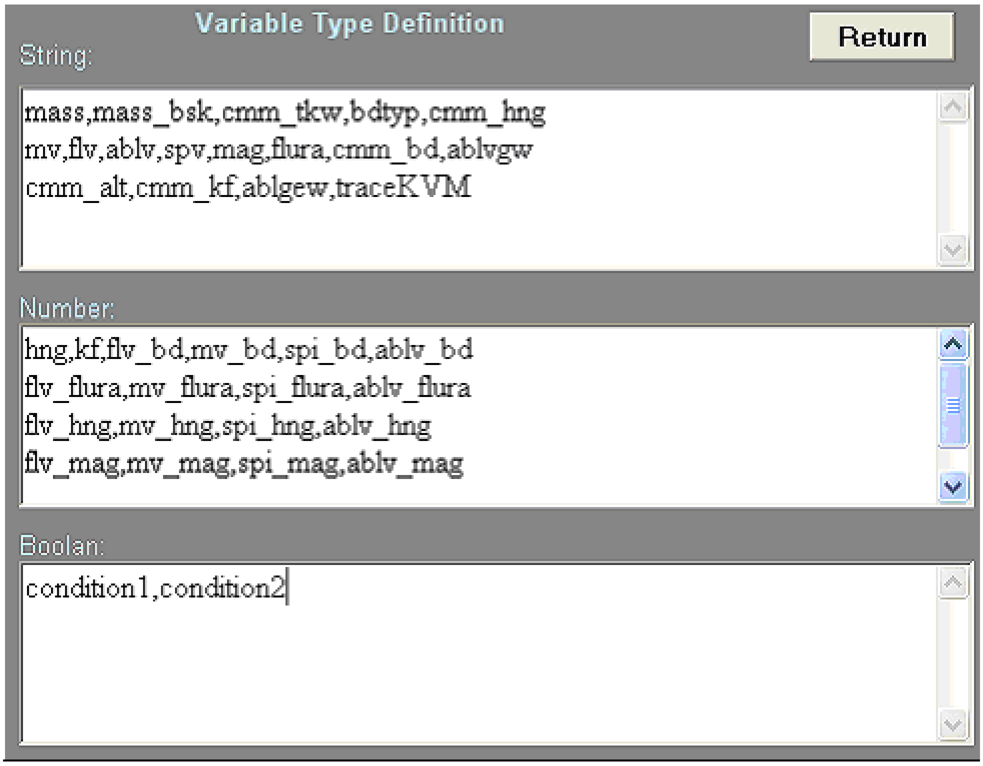
Index of figure

Index of Table

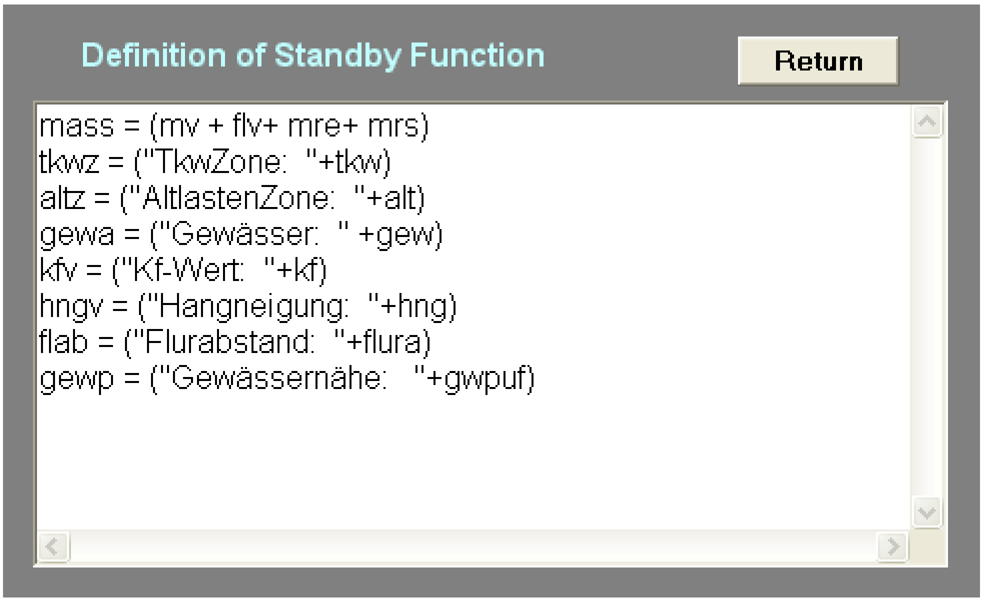
Index of Attachment



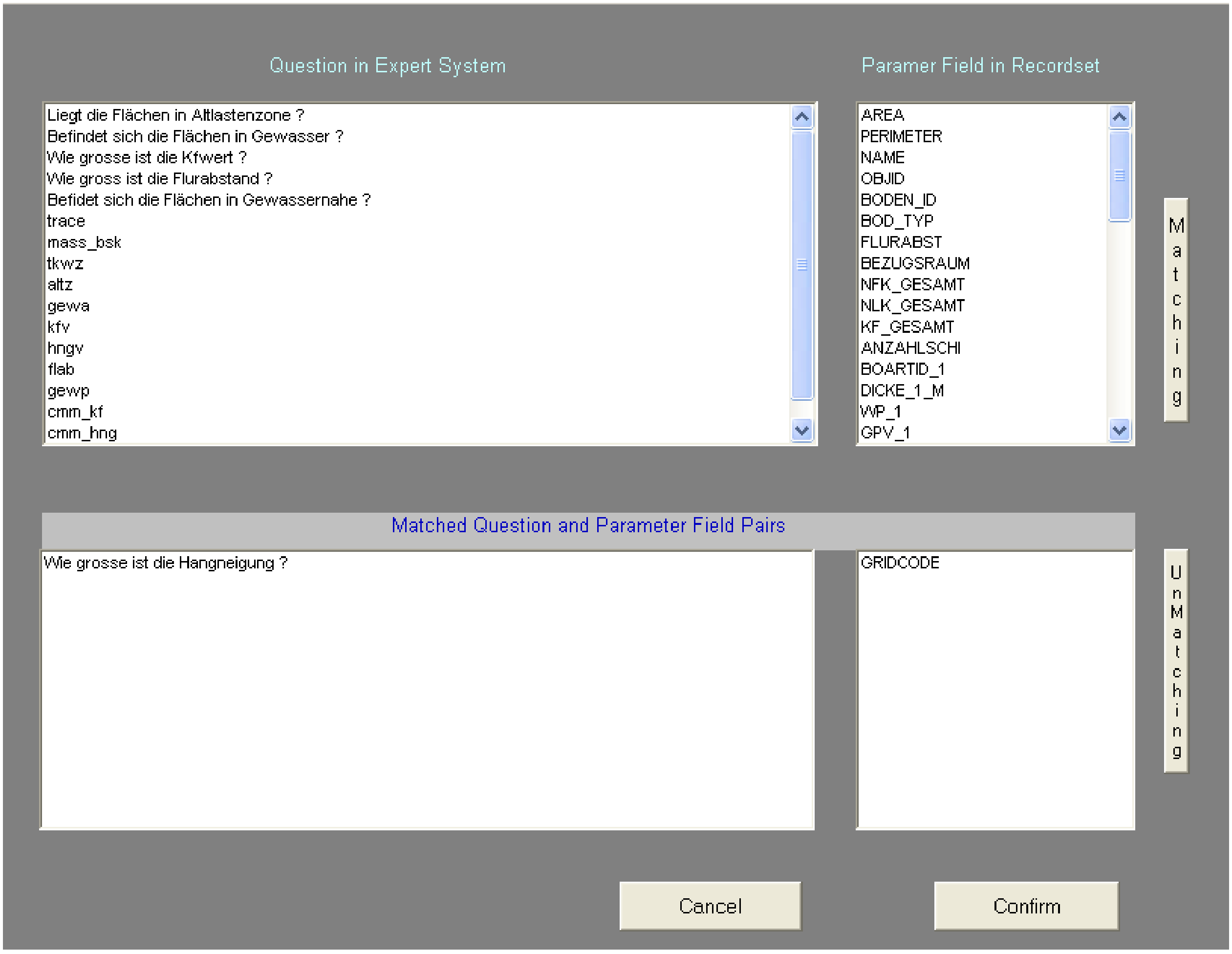
Attachment 1. Schematic concept of a decision tree on selection of on-site stormwater infiltration measures



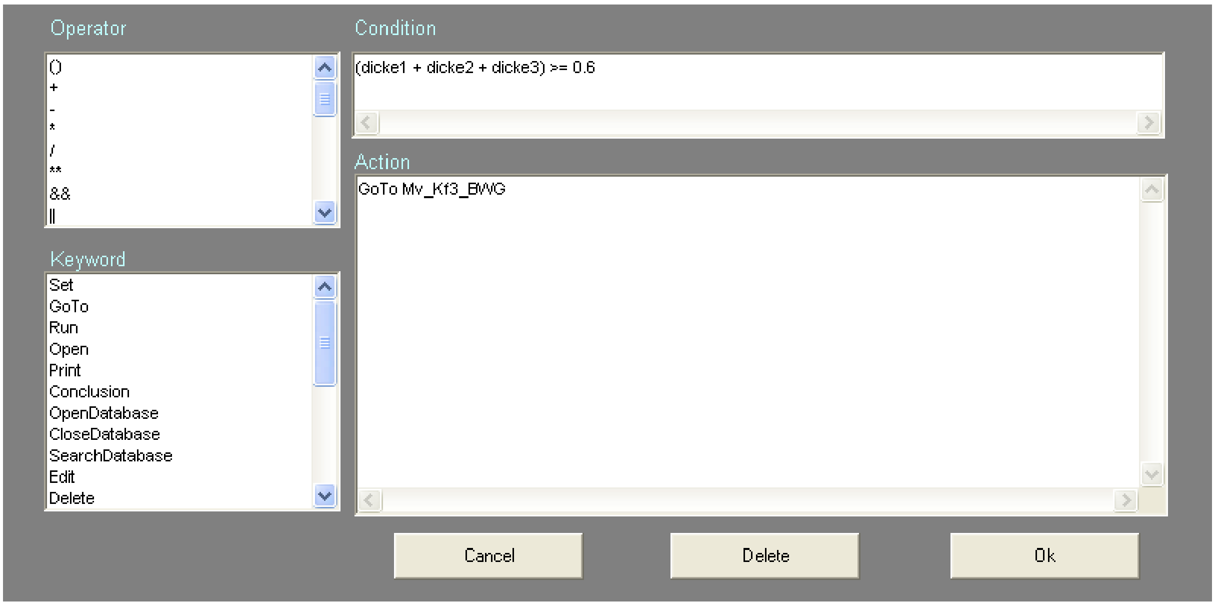
Attachment 2. Interface of assign the type of variables



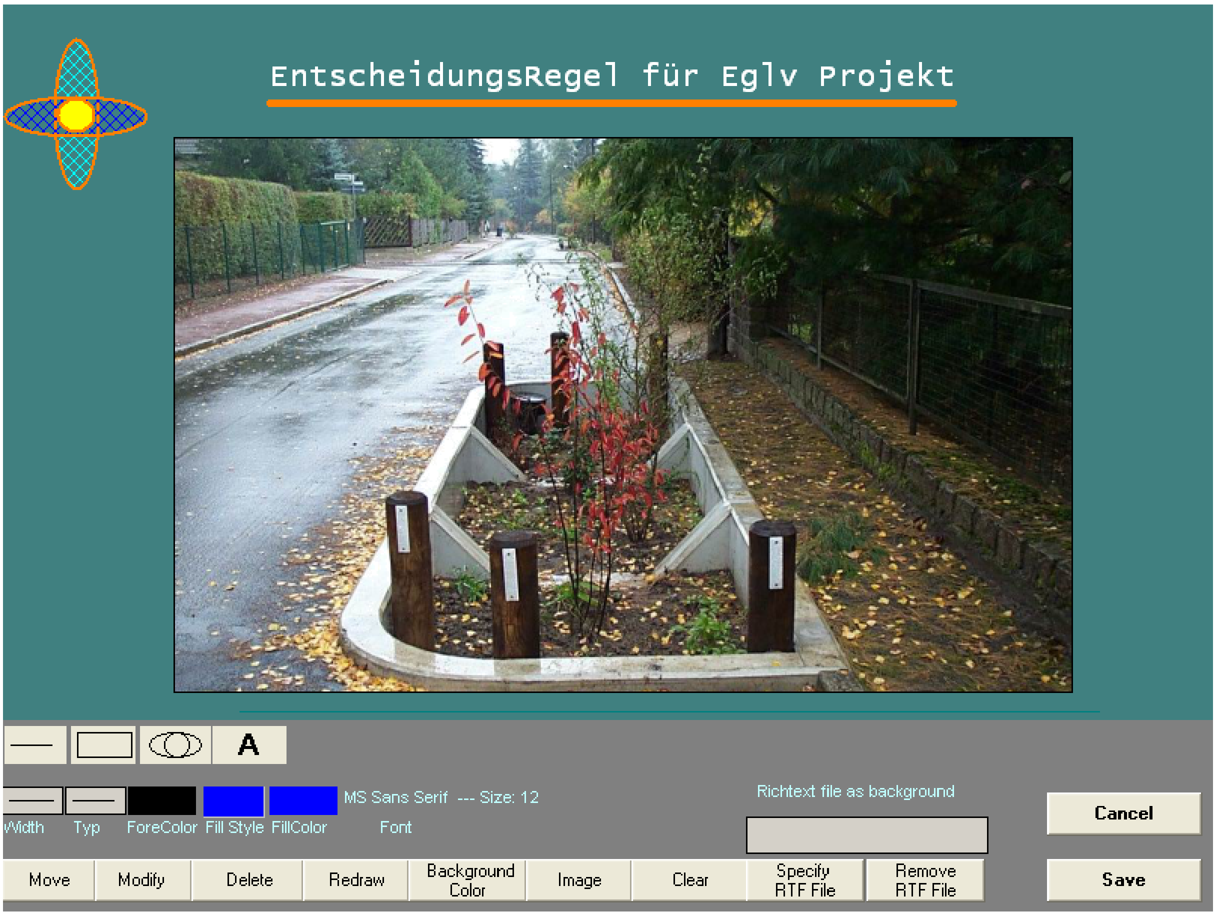
Attachment 3. Interface of definition of functions



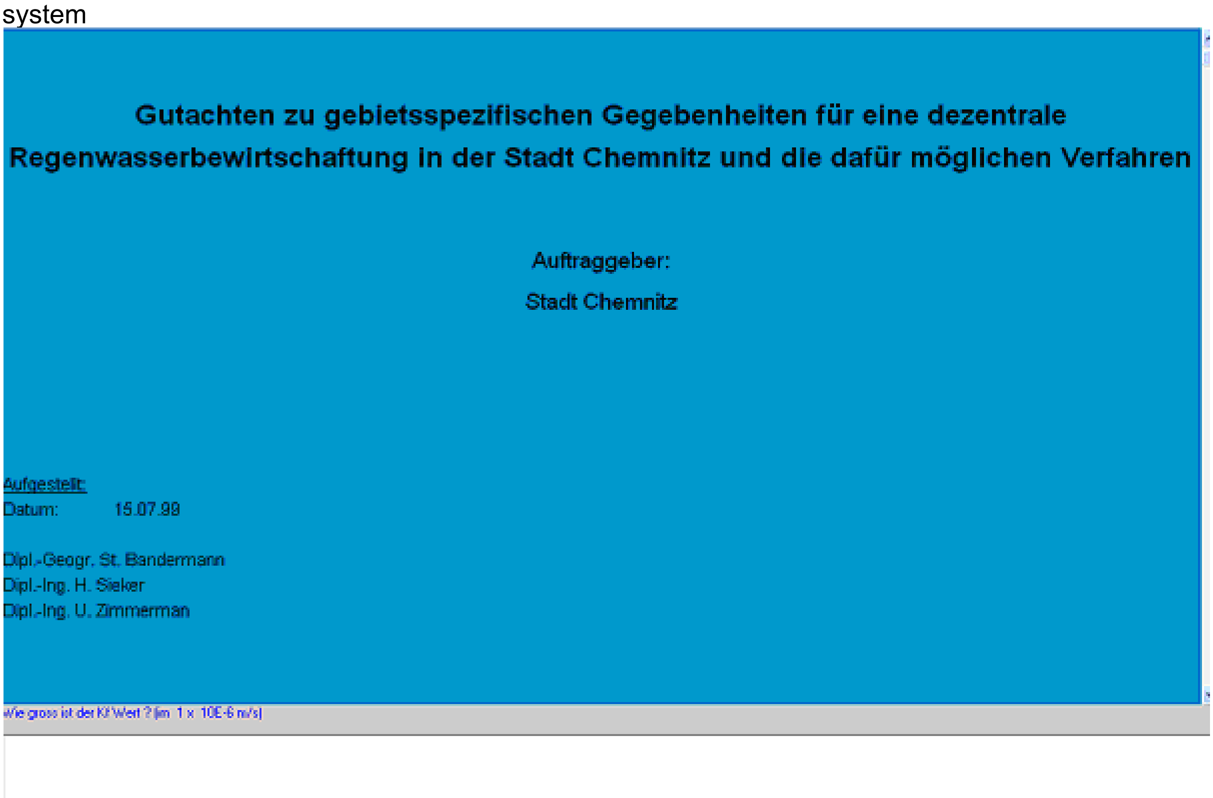
Attachment 4. Interface of matching questions and outputted variables with fields of recordset



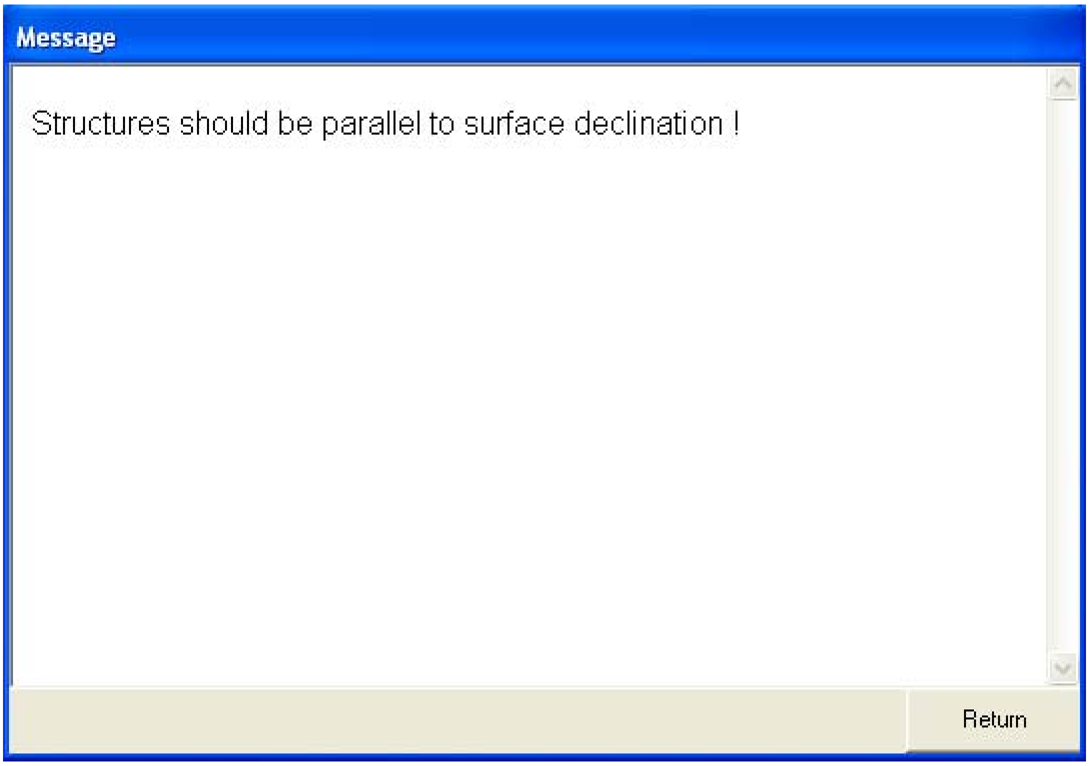
Attachment 5. Interface of formulation a rule (condition and action)



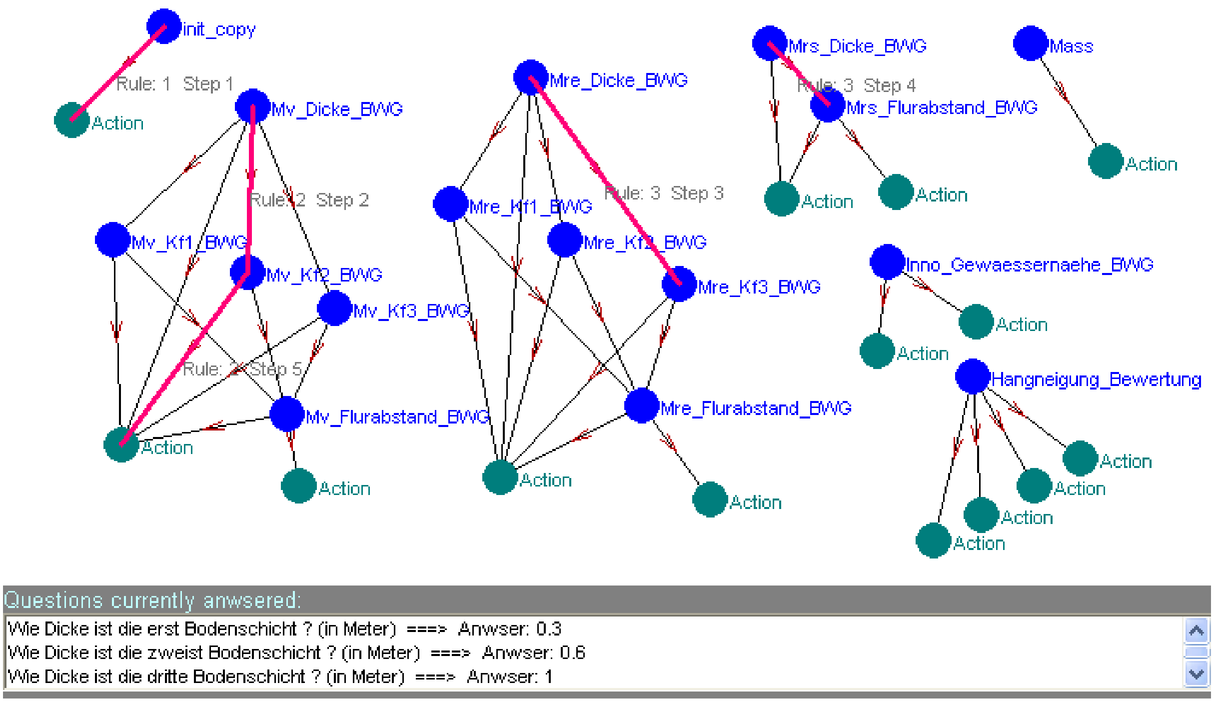
Attachment 6. Interface of designing question background or the face page of a specific expert system



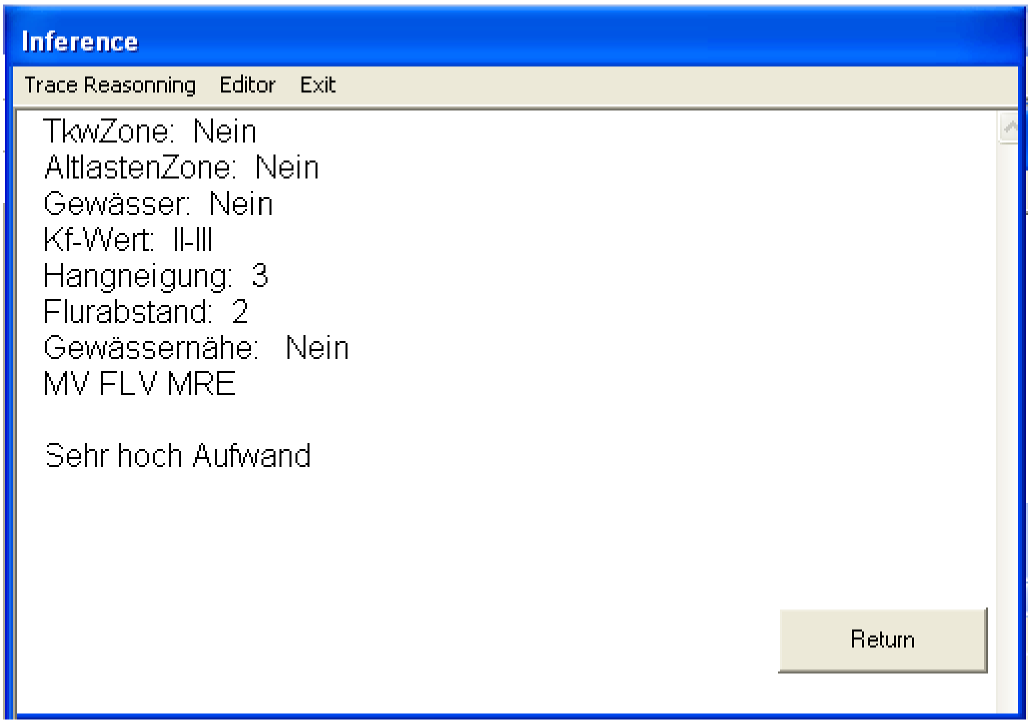
Attachment 7. Rich text file as a question-background at runtime



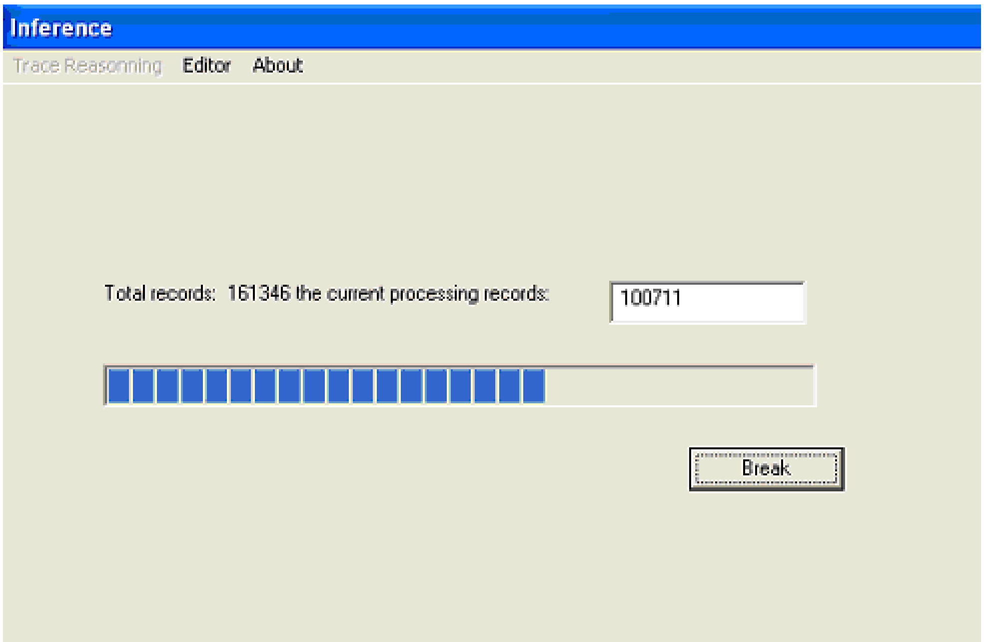
Attachment 8. Interface of message



Attachment 9. Tracing a reasoning process



Attachment 10. Interface of output results



Attachment 11. Interface of execution of an expert system based on GIS data