SewEx – A *flex* Based Expert System for Sewage Treatment Works Support

M.L.E. Dixon†, P.W. Grant‡ and L.G. Moseley‡
‡Department of Computer Science
University of Wales Swansea
Singleton Park, Swansea, SA2 8PP, UK
and
†Hyder IT

Abstract

The development of an expert system for sewage treatment works management *SewEx* is described. This was built with the *flex* expert system toolkit. The system allows the diagnosis of problems in sewage sites which are managed by Welsh Water and accesses the surface assets database over a company network. Other uses of *SewEx*, such as what–if scenarios, maintenance and refurbishment are also outlined. A brief description of a tool, *RuleGen*, is given, which was built to ease the construction of the *flex* rule base. *SewEx* is currently on test at various sewage treatment works throughout parts of Wales and Herefordshire.

1 Introduction

Welsh Water currently operate some 950 sewage treatment works. The day to day task of running such works requires knowledge of chemical, biological, and mechanical principles underlying the treatment process. In addition some of the knowledge is of a practical kind, where experience of a particular site, with all its individual peculiarities, might be important.

This expertise has been built up over many decades and is a valuable company asset. As is common with the expertise in many companies it is dispersed and may not be available at a site when it is needed. Expertise is a fragile asset which may disappear when staff transfer or retire. This is an increasing worry, especially when a company is under pressure to downsize. For these reasons it was desirable to archive this knowledge and make it accessible company wide. Welsh Water were interested in taking one area of expertise – sewage treatment works management – and producing a knowledge base of relevant information, extracted from hard copy records, documents and manuals and most importantly human experts.

The company had no practical experience of expert systems and therefore a collaborative project was instigated under the technology transfer Teaching Company Scheme (TCS), which is part–funded by the DTI. This paper describes the expert system for sewage treatment works management, *SewEx*, which was developed as one deliverable of this TCS project.

There is only a sparse literature on other work related to using expert systems in the water industry. A system called EXTRA is described by Ladiges et al. [1, 2] and seems to have some of the features of *SewEx*. The work of Laukkanen et al. [3] endorses our view that there is need for expert support tools, such as *SewEx*, for intelligent management of sewage treatment works.

2 The problem

In order to show the feasibility of using an expert system for sewage treatment works management a prototype demonstrator system was initially built. This covered only a small fraction of the conditions which arise in practice. It was built in two working weeks, with expertise obtained from only one specialist (a divisional sewage controller).

Below is a typical problem with which an operator might be faced, expressed in the terminology of the water industry:

> A small rural (not large suburban) works is starting to fail on BOD (but not on solids or ammonia). The failure is sudden, but not in storm conditions, and is not a recurring one. It has a screen installed (but no muncher or bypass), has one horizontal primary and one horizontal humus tank. Desludging is done manually twice per week. The operator is fairly experienced at another site, but is new to this one. The input velocity to the works is $1.5m^3$/hour. There is no evidence of any illegal connections. It has an old style distributor (not an open–arm one or an RBC), with an adequate flow, but uneven distribution over the bed. The operator is unsure of the condition of the stilling box on the primary tank, and is also unsure whether the siphon is blowing off, and whether there is adequate biomass, but he does think that the system is occasionally (but not always) overloaded. The humus tank shows no problems, but the primary tank has sludge floating on its surface. There are also dead insects on the bed.

It can be seen that in any formal representation of this description there would be many instantiated variables. Other descriptions of problems might be equally as complex.

From such a problem description, we would expect a management support system to respond with a number of suggestions. For example:

- it would be noted that the operator is inexperienced at this site and that, despite his twice weekly visits and desludging, there is still sludge floating on the primary tank, and additional training for the operator might be suggested;
- measures would be suggested for reducing the input velocity, and recirculation might also be proposed, but only to the outlet of the primary tank;

- at this early stage in diagnosis it is not known whether the BOD failure occurs at the same time as the occasional overloading, but it will be noted that, if they do coincide, a possible cost–effective solution is the installation of a high–rate filter after the primary stage.

We could expect much more than this:

- to include giving advice on checking the data,
- to make suggestions about the effect of any as yet unknown data, etc.

The above is sufficient to illustrate the theoretical complexity of the problems facing even a small prototype system for sewage treatment works management. With the simple example cited here, even excluding variables measured as real numbers, the logically possible number of combinations of problem representation is of the order 10^{10}. It is apparent that any expert system capturing the knowledge to deal with such a variety of problems will be substantial.

3 *SewEx*

After the initial success of the prototype, a full system, called *SewEx*, was planned which dealt with many more situations and covered much of the relevant knowledge which was available throughout the company.

The **initial** purpose of *SewEx* was to take in descriptions of problems arising at a site, and to offer suggestions about the causes of those problems together with possible remedies. This initial brief was comfortably exceeded and once the knowledge base had been constructed, there were all sorts of other uses to which it could be put.

SewEx was developed using *WinProlog* and *flex* [4, 5]. *flex* is an expert system toolkit which fully integrates into Prolog having its own description language, KSL, uses forward and backward (via Prolog) chaining, and features frames and questions and answers. The full flexibility of *flex* was used in the development of *SewEx*. The overall architecture of *SewEx* is shown in Figure 1 (over page).

3.1 Knowledge Acquisition

The knowledge needed was derived from the literature, from process scientists and from operators. About 150 people in the company (out of 3,000 employees) have some of this specialised knowledge in their heads, although no individuals have the breadth of knowledge that the system now exhibits. Interviews were held with 15 people, although only 3 of them were major contributors. This in itself gives an indication of the fragility of the knowledge, and the high potential cost to the company of losing it. The interviews, subsequent knowledge structuring on paper, and the encoding of the initial system in KSL took 5 months of part–time work. As reported by other researchers, Knowledge Acquisition is always a time–consuming and difficult activity [6, 7] and there is still a need for good tools to aid in this activity [8].

Further details of the knowledge acquisition phase of the project can be found in [9].

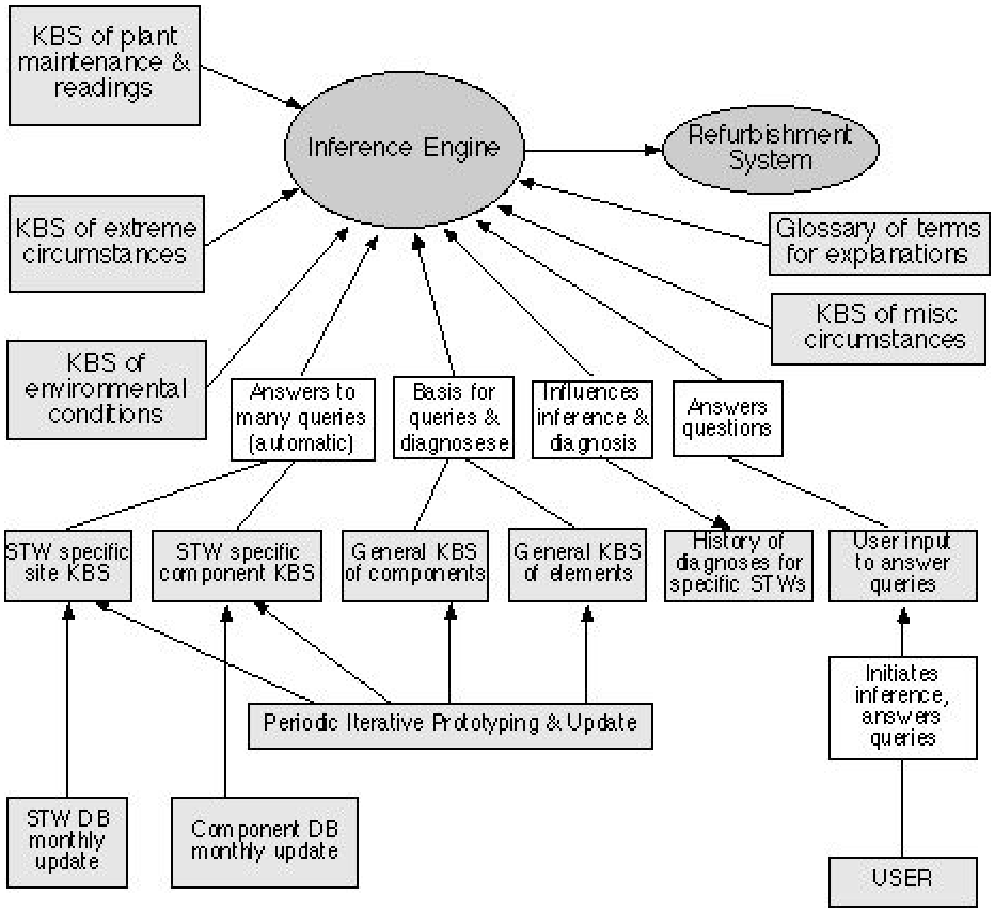

Figure 1: General Architecture of the KBS

Additional development to *SewEx* (including links to the company network, integration with asset databases, undertaking validation, and initial testing) took a further 12 months of part–time work. It is estimated that the total human effort was approximately 7-8 person–months of work, albeit highly skilled work.

3.2 Design Issues

There are two major elements to the expertise – rules and facts.

3.2.1 Factual Knowledge

The facts knowledge base contains assertions about the sort of plants that exist, details of components and particular details of actual sites. Information has to be known about which elements go to make up a component; and for each element, which are the relevant parameters (both in general and for a particular site). One of the first tasks was, therefore, to draw up a typology of the various parts and sub–parts which are needed for reasoning about sewage treatment, and to specify the relevant parameters.

Four main types of plant were identified

1. Filtration plants
2. Aeration plants
3. Package plants
4. Septic tanks

Each plant, depending on its type, will contain a variety of components. 29 components were identified for inclusion in the system. Examples appear in Table 1.

Inlet	Pump station	Macerator	Primary settlement tank
Balancing tank	Siphon chamber	Filter distribution (1st stage)	Oxidation ditch
Outlet	Storm tank	Sludge holding tank	Dosing chamber
RBC	Humus tank	Clarifier	

Table 1: Examples of Components

In turn each component contains *elements* which can be specific to individual components, or generic. Table 2 exhibits some of these elements, there being 36 in total.

Pipes	Channels	Weirs	Valves
Pumps	Scum-boards	Sparge arms	Siphon pipes
Filter media	Microbes	Onion bags	Auto greaser
Fat trap	Grit pumps	Waterwheel	Hydrostatic head

Table 2: Examples of Elements

Finally each element would have one or more of 23 identified *parameters*, some of which we list in Table 3.

Material type	Capacity	Flow rate	Depth
Alarm level (min)	Max hours run	Retention time	Pipe diameter
Pressure	Storm capacity	Year purchased	% treated flow

Table 3: Examples of Parameters

The other objects identified included:

- Component Connectors (7 types),
- Monitors and Detectors (8 types),
- Environmental Circumstances (4 sub–types),
- and Miscellaneous Circumstances (5 types).

This typology provided a static model for sites and can naturally be organised as hierarchical knowledge. All the treatment works which were to be included in the system could be described using the objects identified above. For a particular site, of course, one needed to state its type and the components which it contained. *SewEx* allows the user to present a site layout plan graphically, with each of the components displayed, together with the connections between them (see Figure 2).

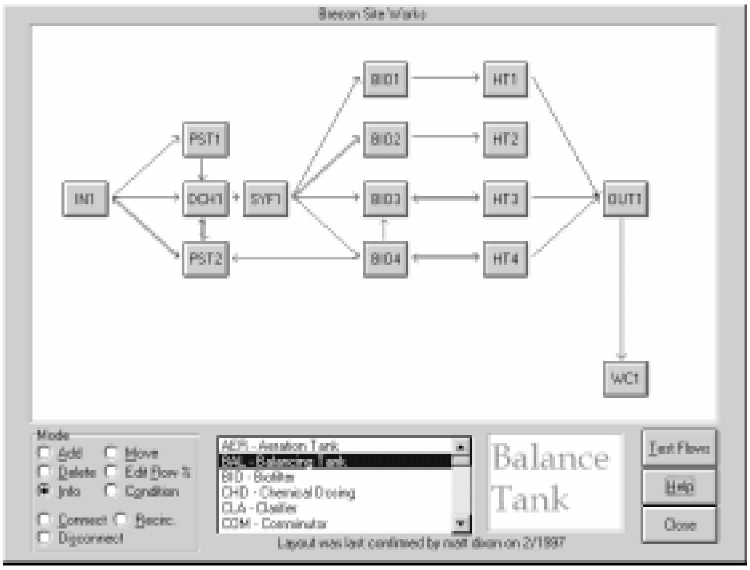

Figure 2: Site Layout Diagram

Each site has to be populated with the data about the parameters of each component (capacity, flow–rate, age, etc.). The facts can come from one or more of three sources:

- Routinely recorded company data (about components, sites, etc.)
- A special database, unique to *SewEx*, which is fuller than the company data.
- User provided data about the current problem at a given site.

The apparently simple step of providing a way of merely describing what a particular works contains turned out to be more complicated than one had at first imagined. The full specification of these factual items constituted a 75–page document. Now there is a general method for describing any site very specifically. It is clearly important that these factual background data are entered accurately for the reasoning to yield correct results.

3.2.2 Rule Based Knowledge

Once the factual knowledge base of a site had been established the next step was to elicit the diagnostic **if–then** rules from the various sources (usually from experts). In *SewEx*, the rules were to be expressed in the form of clauses in Prolog and *flex*. To permit the easy and consistent encoding of rules, a special program, called RuleGen was written [10]. As there were over 1,400 rules in the final system, many of which interact with each other, there had to be some way of presenting them (other than in a simple text format) so that human users could comprehend them.

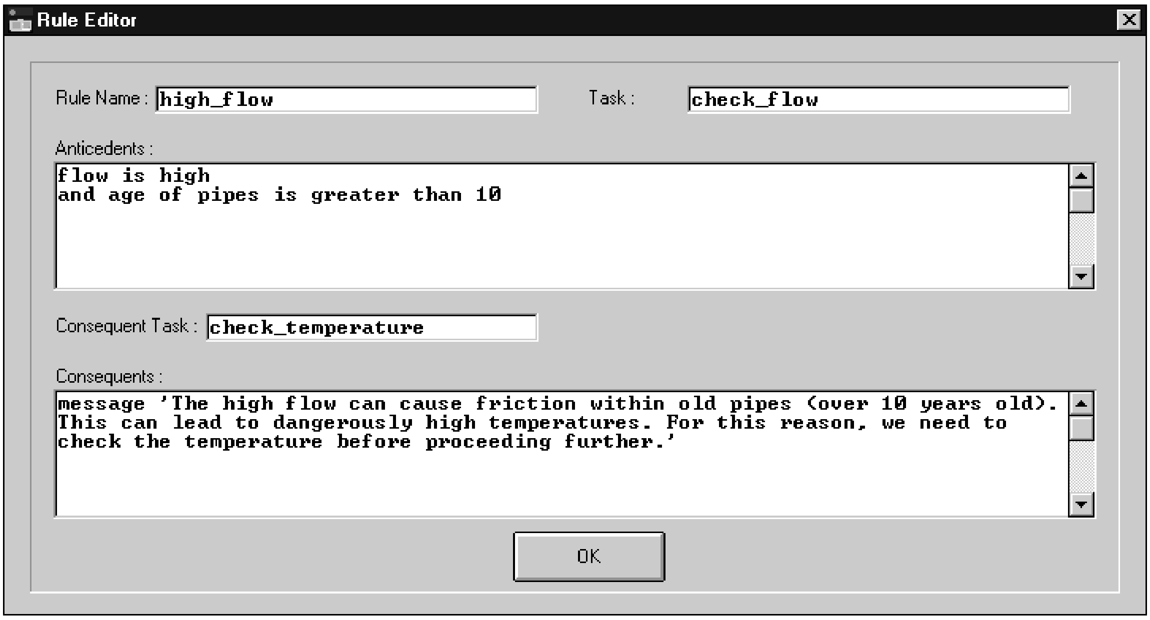

Figure 3: RuleGen Editor Screen

Figure 3 shows the screen for entering and editing the following textual rule:

```
IF  task is check_flow
    AND flow is high
    AND age of pipes is > 10
THEN
    print the message
    AND task becomes check_temperature
```

Changing the current task is used to control the firing of rules. *RuleGen* permits the display of rules and their linkages in the form of a graphical tree. The user can add and delete rules, move them around, and insert and remove links between them. The rule tree display from *RuleGen* is illustrated in Figure 4

To give a feel to the complexity of some of the rules, Figure 5 illustrates part of the rule tree for inferring why a biofilter is dying.

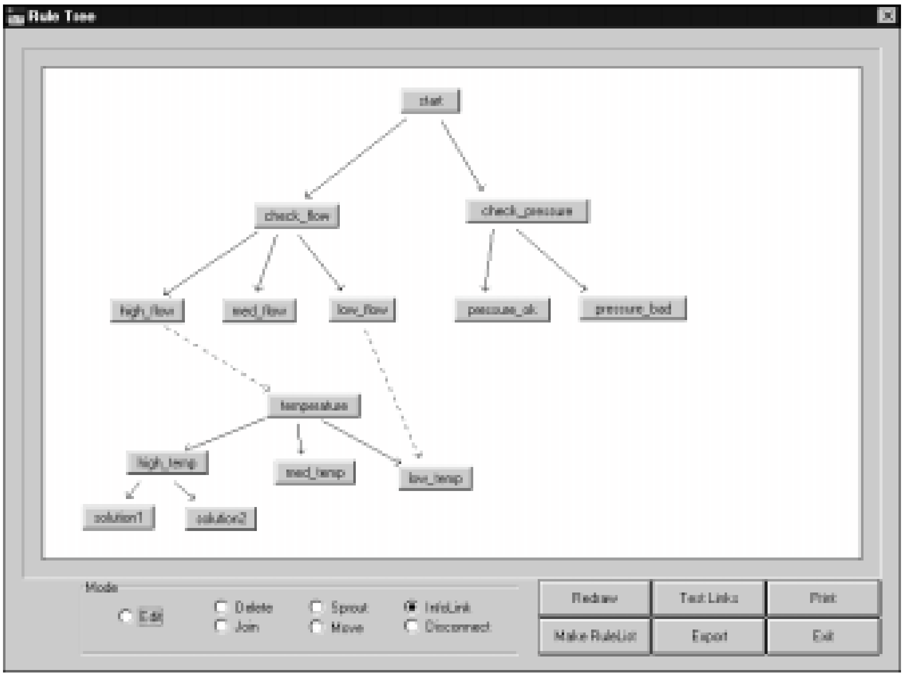

Figure 4: RuleGen Rule Tree Screen

3.3 Deployment of the System

Welsh Water has a network with over 3,000 connections and *SewEx* is available from any point on that network. Updates of the program can also be installed remotely from the network. An individual user may have locally on his or her workstation details of only a few sites of local interest. However, the system is linked to the company's Surface Assets Database (SADB), and details of other sites can be downloaded when needed. The interface is a typical Windows one, developed using features offered by WinProlog.

An individual user can:

- add or delete sites from their local computer,
- find sites with similar characteristics,
- produce a report on the current condition of the site assets,
- get general information about the site (e.g. any problems with access or security),
- do what–if testing (e.g. by altering connections between components to see what the effects would be),
- obtain a checklist of maintenance requirements for a given site,

as well as undertaking the main task of the system – the diagnosis of a specified problem.

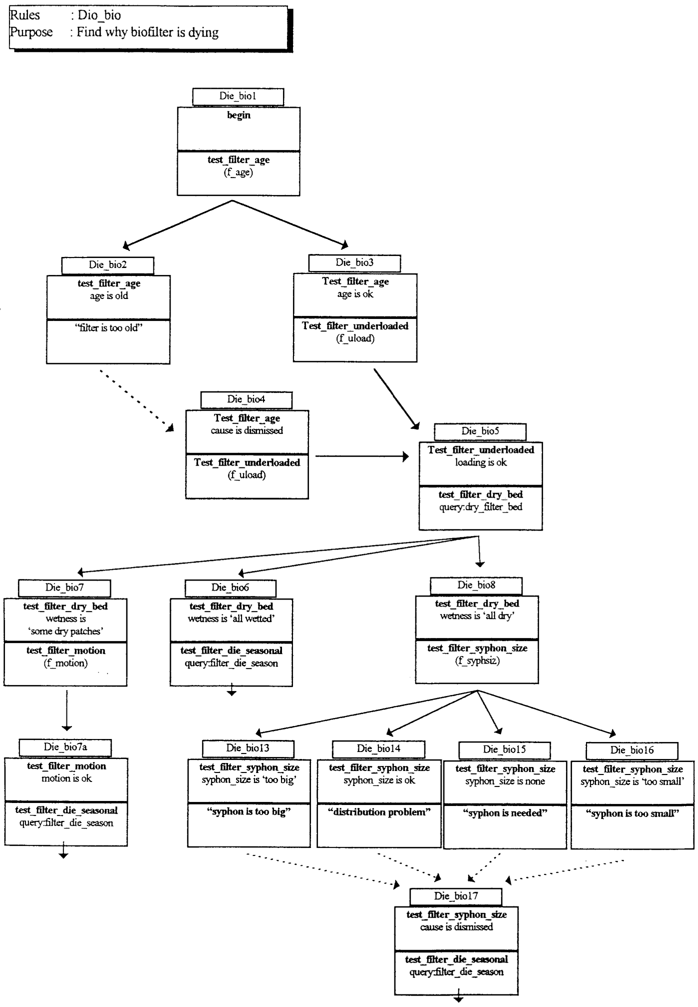

Figure 5: Part of rule tree for biofilter dying

From the layout screen, Figure 2, one can edit the configuration of the works, for example, by adding or deleting components, changing the connections between them, modifying flow rates between them, etc. This means that the what–if testing can be undertaken graphically.

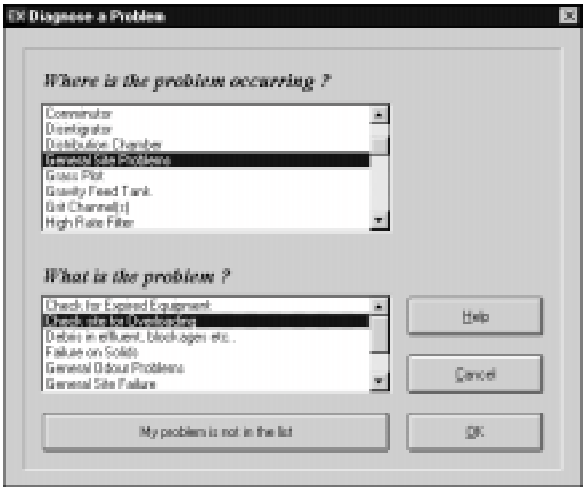

Figure 6: Diagnosing a problem (Input Screen)

For some parameters, the user may not have detailed information readily available at the time of the consultation. The system therefore permits the user to make an estimate if he so wishes. The main diagnosis input screen is illustrated in Figure 6.

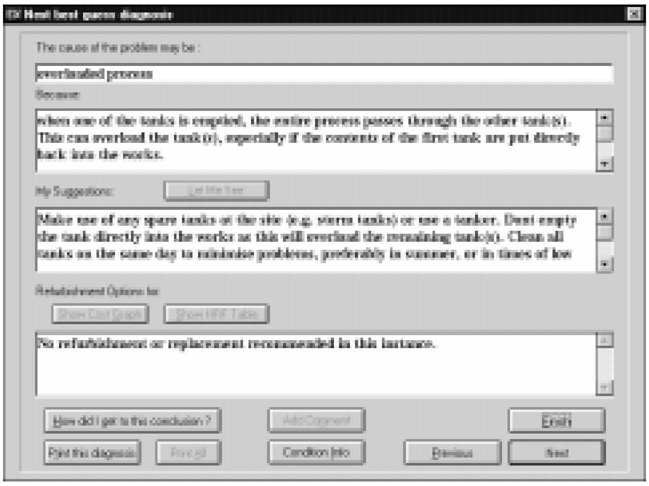

Figure 7: Diagnosing a problem (Output Screen)

SewEx will continue to draw data from one of the databases, or ask the user questions, until it has narrowed down the possible problems to a manageable list. It will then move on to offer one or more diagnoses. Note that the system has a concept of best, second–best, etc. diagnosis (see Figure 7). The variety of buttons on this screen demonstrates additional functionality, e.g. displaying the line of reasoning used to infer this conclusion.

4 Uses of the System

The company has substantial and expensive policies for attempting to gather and maintain data about its physical assets. There is a large programme in operation to record (and to update regularly) details of its assets, both above ground and sub-surface. This matters, as such assets have a value of over 10 billion with, for example, over 10,000 pumps in use. When it came to documenting the knowledge which controls these assets, the situation was less well developed. Although many operations manuals had been produced, until recently expertise was not seen as a recordable, maintainable or machine-processable asset.

A major effect on company culture has been that such a recognition has occurred. Sewex is now a major plank in the company's asset management plan. Effectively, knowledge management has been introduced into the company. When one has recorded (for whatever purpose) in machine-processable form not only the data, but also the static and dynamic reasoning knowledge, all sorts of other activities become possible. Examples from Sewex include:

- investigating what–if scenarios;
- producing asset condition reports;
- prioritising maintenance and refurbishment;
- enhancing quality;
- training;
- making projections and
- obtaining feedback.

Each of these is discussed in a little more detail in this paper. Complete descriptions can be found in [11, 12]. Figure 8 shows the main start–up screen.

Figure 8: Main Introductory Screen

4.1 Investigating what–if scenarios

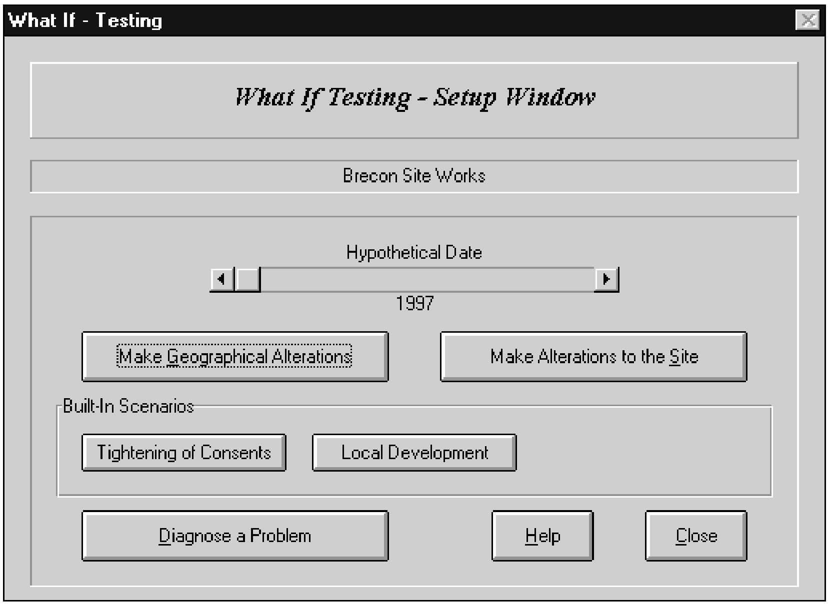

Figure 9: What–if Screen

The system includes not only rules, but a number of standard hydrological equations, and a substantial amount of data. The user can investigate many scenarios (Figure 9):

- the effect of an increase on population on the works as it stands;
- the effect of proposed local industrial or housing development;
- the effect of adding or changing one or more components;
- the effect of recirculation (done graphically on–screen), etc.

Diagnoses can be run to see the above effects and obtain prior warning of future problems. The changes are effective only during the what–if phase.

4.2 Producing asset condition reports

This is obvious, and permits an operator or manager to obtain, at the press of a button, a report on the state of repair, age, previous problems of the components installed at a site, produce a site–relevant maintenance check–list, etc.

4.3 Prioritising maintenance and refurbishment

As the system includes cost and pricing information, it can answer questions such as "Which are the most urgent items for upgrading at this site, given a maximum budget of £X ?" It can list repairs or refurbishments required by cost and by degree of urgency. Indeed, if it cannot find a simple cost–effective solution to a problem which it has diagnosed, it will automatically invoke the refurbishment sub–system to ascertain whether a more substantial investment is needed to prevent the re–occurrence of the condition.

4.4 Quality enhancement

The system will suggest, if requested, desirable quality enhancements in line with company policy. Its suggestions will attempt to reduce the number of methods of refurbishment, permitting standardisation, bulk–purchasing, and other cost savings to be achieved. So, if the suggestion is to install a high–rate filter, the system can display a table of best–buys, or a graph of the costs of different purchase options for different population sizes served by various works.

4.5 Training

Many of the questions may use terminology which is unfamiliar to a particular user. In such cases, the provision of an *explain* button means that the user can receive an explanation and thus some training. For example, if the system asks "What is the percolation value at this site?" and the user asks for an explanation, the system will respond with instructions about digging a hole, the dimensions of the hole, how long to let the water stand, how to measure the length of time which it takes to drain, etc. The fact that the system can also explain the line of reasoning which led it to a given conclusion is also a useful training aid. Explanations provided often include photographs or other graphics which further helps to make this training function a very practical one.

4.6 Projections

The system can make predictions about the likely future condition of the site, with the user specifying the time period for the projection. This is an useful management tool.

4.7 Feedback

There are a variety of feedback mechanisms. Perhaps the most useful one is the facility for a user to add, at the end of the consultation, a free–text comment. This is recorded on disk, and is available to the system developer when the time comes to revise and modify the rules and helps keep the knowledge base up to date.

Much of the data comes from the SADB, which is a trusted source and is accessible on–line over the company's network. If the user notes that the data from this source is not currently correct, this information can be fed back to the database administrator, for later updating. This maintains central control of data quality, and ensures that all consultations are undertaken using the best currently available data.

4.8 Other Developments

The success of Sewex has led to a realisation that knowledge is an asset, and that software exists or can be developed to exploit it. This is now manifesting itself in a wide variety of ways. Several machine learning investigations have been undertaken, with measurable economic benefits. Other expert systems are under development. Two with which we are closely associated are (a) a case based reasoning system for water treatment advice (b) a financial system. The general point which we would stress is that once you have knowledge, you can use it for many purposes which were not envisaged in your original plans.

5 Summary

This project has shown that for a large–scale problem, the expert systems approach of gathering, structuring, and linking knowledge and inferencing methods is feasible in a reasonably short time–scale. More importantly, it has shown convincingly that the codifying of knowledge and reasoning (which has to be undertaken to produce an effective diagnostic system) can then produce additional benefits. Sophisticated what–if scenarios can be undertaken far more simply and powerfully than is possible in, for example, a spreadsheet. The added value of producing asset condition reports, prioritising maintenance expenditure, making projections, giving suggestions for quality enhancements, helping with training, and ensuring the provision of reliable data make the development of this program highly cost–effective. The fact that it can do so within strong regulatory constraints is yet another benefit.

Acknowledgements This work was supported by the Teaching Company Directorate/DTI and Welsh Water under grant GR/K64112. We would also like to thank Nick Humphrey, Phil Pemberton, Chris Murtagh and other colleagues at Lusis Business Solutions Ltd. and Welsh Water, who have provided input into this work.

References

[1] F. Ladiges and R. Kayser. Applied off-line expert systems for effluent, operational and technical problems of waste water treatment plants. *Water Science and Technology*, 30(2):157–164, 1994.

[2] F. Ladiges and A. Mennerich. Application and experience with expert systems for operation of waste water treatment plants. *Water Science and Technology*, 33(12):256–268, 1996.

[3] R. Laukkanen and J. Pursiainen. Rule-based expert systems in control of waste water treatment systems. *Water Science and Technology*, 24:299–306, 1991.

[4] P. Vasey, D Westwood, and Nick Johns. *flex Expert System Toolkit*. Logic Programming Associates Ltd., 1996.

[5] B.D. Steel. *LPA Win–Prolog Programming Guide*. Logic Programming Associates Ltd., 1995.

[6] J. Durkin. *Expert Systems*. Macmillan Publishing Company, 1994.

[7] B. Gaines and J. Boose, editors. *Knowledge Acquisition for Knowledge-Based Systems*, volume 1 of *Knowledge-Based Systems*. Academic Press, 1988.

[8] J. Boose and B. Gaines, editors. *Knowledge Acquisition Tools for Expert Systems*, volume 2 of *Knowledge-Based Systems*. Academic Press, 1988.

[9] M.L.E. Dixon. Investigating small sewage treatment works. Technical Report TCS97.1, Lusis Business Solutions Ltd., 1997.

[10] M.L.E. Dixon. Rulegen, the rule generator. Technical Report TCS97.2, Lusis Business Solutions Ltd., 1997.

[11] M.L.E. Dixon. Sewex – design of the knowledge base and objects. Technical Report TCS97.3, Lusis Business Solutions Ltd., 1997.

[12] M.L.E. Dixon. Sewex – user guide. Technical Report TCS97.4, Lusis Business Solutions Ltd., 1997.