Decision Support for Sentencing in a Common Law Jurisdiction

N.Hutton A.Patterson
C.Tata
Centre for Law, Computers and Technology, The Law School, University of Strathclyde,
Glasgow G4 0RQ, UK
e-mail:{neil,ALAN,CYRUS}@law.strath.ac.uk

J.Wilson
Department of Computer Science, University of Strathclyde, Glasgow G1 1XH, UK.
e-mail:jnw@cs.strath.ac.uk

Abstract

The inherent variability of each new conviction and the volume of potentially useful data complicate the provision of relevant information to judges carrying out sentencing activities. This paper describes a prototype sentencing information system which allows a judge to retrieve previous examples of sentencing behaviour by interactive instantiation of detailed templates. The case-based advisory system uses a direct manipulation interface to facilitate user interaction. The direct manipulation interface was well received and judges found that they were rapidly able to construct meaningful queries and retrieve appropriate data despite having limited prior experience of computer use. We identify the importance of dynamic query systems in relation to further development of applications of this type.

Keywords: sentencing information systems, case-based advisory systems, direct manipulation interfaces, decision support systems.

1 INTRODUCTION

1.1 Organisational Decision Support Systems

The advent of changes in the environment in which large organisations operate has forced many to re-orientate their efforts. The use of software such as distributed decision support systems is fundamentally important in the practice of business process reengineering [1] . Computer assistance for decision support systems requires the use of tools for searching and retrieving data, viewing and manipulating the results of such a search, analysing and organising the data and inserting new data. In the context of business information processing the technical focus of decision support systems has moved from large centralised data processing applications to distributed inter-organisational information systems. There has been a corresponding increase in the user population which is now more likely to have little opportunity to develop traditional computer skills. Added to this is the enhanced likelihood that users will have limited perception of the semantics of data stored within such systems [2].

The need to respond to changes in the environment is also being experienced in organisations which control the administration of justice. In common with the impact of such changes on commercial organisations, there is considerable interest in the use of computer systems to enable evolutionary changes to preserve the best features of pre-existing practices. The development of software which provides judges with information about previous sentencing behaviour and at the same time preserves judicial independence is an example of the provision of such software support.

1.2 Sentencing Information Systems

Computer support for sentencing has been investigated in a number of jurisdictions. Chan [3] describes a system implemented in New South Wales which provides access to first instance sentences as well as to sentencing law and appeal court judgements. The system is implemented in a mainly statutory jurisdiction and convictions are entered by specifying the act and section which relate to the offence. Characteristics of the offender may be specified by the user and sentencing patterns are displayed to show the distribution of particular types of sentence. The system is menu driven and the user chooses options by use of the cursor control keys and the keyboard. The data is centralised and can be accessed over a dial-up link from remote sites.

A similar approach to decision support for sentencing has also been investigated in Canada [4]. This approach uses broad characteristics to classify convictions on the basis of the seriousness of the offence, the culpability of the offender, the offender's previous history, aggravating and mitigating factors, the effect on the victim and the frequency of the offence in the community. The system constructs a table showing the distribution of sentences of various types. Although this system was supplied to judges, it was found to be of limited utility since judges were not motivated to use information in this form.

An alternative approach to providing computer support for sentencing is to use a knowledge-based system to guide sentencing decisions. Murbach and Nonn [5, 6] describe a prototype sentencing decision support system which is limited to fraud cases. Discussion with the judiciary was used to establish a sentencing model and subsequently construct a knowledge-based system to incorporate the reasoning process.

The knowledge-based systems approach is particularly relevant where sentencing guidelines are used to inform sentencing practice. ASSYST [7] was developed to provide such support. The system reflects the Sentencing Reform Act 1984 and uses an heuristic approach to combine offence seriousness with previous conviction history to produce a guideline sentence. This system has been successfully implemented and is in use throughout the Federal criminal justice system. Bainbridge [8] describes a knowledge-based system which addresses the question of the legal competence of a sentence. The system incorporates details of sentencing principles for robbery and theft. The JUDGE system [9] uses a case-based reasoning approach to suggest appropriate sentences for a limited range of crimes on the basis of a variety of offence and offender characteristics.

1.3 Case-based systems (CBS)

Case data can be used to support decision making by providing systems with a range of reasoning capabilities. The case-based reasoning model (CBR) [10, 11] provides for automatic generation of problem solutions whereas in the case-based advisory (CBA) approach, the aim is to support the human problem solver [11]. There is a considerable area of overlap between these two models.

CBR is one of the Artificial Intelligence technologies which has successfully translated from research effort into useful applications. It holds a position equivalent to rule-based systems, although the implicit meaning of the term 'expert' differs between these paradigms. In CBR systems, the expertise is focussed on skill at recalling previous experiences which are relevant to current problems. In rule-based systems the work of the expert is simulated by applying generalised heuristics to a set of facts which result in a new deduction.

CBR systems operate in a series of steps:

- specification of a problem in a form which is related to the specification of other similar problems,
- retrieval of prior cases which match the problem specification,
- selection of those cases with the best precision.
- adaptation of cases to match a new set of circumstances,
- evaluating the solution proposed by the adaptation stage,
- addition of new cases to the case base.

The output of a consultation with a CBR system is the classification of a new case. In this context, it is important to retain cases in which the outcome was viewed as valid as well as contrary cases where the outcome was invalid.

Applications of CBR can be categorised on the basis of the purpose for which the case-base is employed. Two patterns of use are recognisable: problem-solving and interpretive. In problem-solving systems, the aim is to generate the solution to a new problem by using the solution from an older, similar problem. The adaptation stage is of major importance in producing a satisfactory solution since it is likely that no exactly matching cases will be found in the case-base. This approach is typically used in the application of CBR to generating solutions to design problems.

The interpretive approach to CBR focuses on the use of previous cases to provide a context against which a new case can be evaluated. There is less emphasis on the need for the system to produce a solution and greater emphasis on the need to arrange that case retrieval provides a set of cases which can be used to put current problems into a perspective. This style of CBR is typical of the way in which a lawyer would use precedent as support for current arguments. The match between interpretive CBR and the information requirements of the legal profession has long been recognised.

In both problem-solving and interpretive CBR systems, the stages concerned with problem specification and case retrieval are of major importance. The first of these stages can be implemented by the instantiation of a template to represent the structure of the current problem and the matching of this template in the case-base.

Two significantly different approaches to template instantiation exist : user-directed approaches and techniques of providing assistance usually by the use of rule-based systems. User-directed template instantiation is the simplest of the approaches and has been used to collect relevant cases where the domain is clearly defined [12]. In this approach, the user is prompted to specify the predicates to be used in case retrieval.

Where the case predicates are of an abstract nature, several authors have applied rule-based systems to expand the definition. This approach to integrating rule-based systems and CBR is appropriate to legal case-based systems where predicates are imprecise or "open textured". Thus a case-base may contain data which relates implicitly to the predicate "malice". In order to retrieve such cases this predicate would require to be mapped to predicates explicitly stored in the case-base [13, 14].

CBA is an alternative approach to case-based systems described by Kolodner [11]. This pattern allows the system less autonomy and relys on the user to provide adaptation and associated decision making. CBR systems provide a software model of human reasoning whereas by contrast, case-based advisory systems are used to give direct support for human reasoning. The system is specialised to provide partial matching to ensure that a range of potentially useful cases are retrieved.

Case retrieval can involve alternative approaches to the indexing process which is necessary for matching the template to the data in the case-base. Indexing can be carried out on the basis of surface features or structural features. The use of surface features is perhaps the most common pattern in applications of case-based reasoning. Such surface features can be incorporated in the template to generate a database query in a language such as SQL (Structured Query Language). This approach may be enhanced by weighting attributes and combining weights to arrive at a measurement of nearness, thereby giving a means of ranking the potential relevance of cases retrieved.

The process of structural indexing is similar to the construction of a decision tree which is used to classify cases. The process is related to machine learning by induction but leaf nodes in the decision tree contain groups of cases rather than single instances [15].

1.4 Direct manipulation interfaces

Shneiderman [17] first characterised direct manipulation interfaces (DMI) following work at Xerox PARC. The features of such systems are that objects of interest are represented continually on the computer display and manipulation of these objects is achieved by physical actions such as pressing

labelled buttons, dragging articles on the screen etc. Interactions between the user and the system are characterised as being rapid, incremental, reversible and having immediately visible impact.

The user of such DMIs is empowered to concentrate on the semantics of the interaction. This contrasts with conversational interfaces where the user needs to grasp both the syntax and the semantics of the system. Examples of such conversational interfaces are database query languages such as SQL. The philosophy for direct manipulation interfaces is founded on the assumption that users are likely to be naïve and to be only occasional users of such systems. The typical 'desk-top' metaphor of Macintosh and Windows applications requires limited user support documentation, relying instead on the assumption that the user will be able to explore the system interactively and that where such exploration fails, online help facilities will be enough to provide the necessary assistance.

Empirical evidence suggests that in database manipulation exercises novice users find DMIs faster, easier, less error prone and more stimulating than equivalent conversational interfaces. Where tasks become more complex or where there is a need to specify repeated operations on data objects, conversational interfaces may well be more appropriate [18].

2 The SIS project

The aim of the Sentencing Information System (SIS) project is to produce an application which allows a judge to retrieve previous examples of sentencing behaviour. Such examples are selected by the system as being closely related to a conviction of current interest. The fundamental importance of involving the whole judiciary in the project was recognised at an early stage. We also recognise that the system must be 'transparent' to the judge - the judge must be in complete control of the specification and retrieval of the cases. This requirement implies that user-directed template construction and unweighted indexing of surface features are the appropriate methods of case retrieval. Although user directed template construction is appropriate in this prototype, it can involve extensive user interaction In this context it is clear that direct manipulation interfaces provide considerable advantages over narrative, keyboard-driven interfaces.

Since the work pattern of a High Court judge in Scotland can involve travel to any of the locations at which the High Court may sit and the process of sentencing is carried out relatively infrequently, a pattern of operation is required which combines maximum portability with support and ease of use.

We considered that rule-based systems technology was inappropriate in this exercise because of the potential difficulty in extracting a set of rules adequately to describe the complexity of the domain. Furthermore, the rule-based idiom could be interpreted as compromising judicial independence by suggesting a "correct" sentence rather than acting as an aid to recollection by retrieving a set of comparable cases. The case-based advisory approach satisfactorily avoids both of these difficulties whilst at the same time leaving the decision on appropriate sentence in the hands of the user. The data required to underpin the system was available in the form of indictments and case papers.

2.1 Systems Design

During a consultation of the SIS a judge may be interested in, for example, the sentencing patterns for assaults. S/he may be particularly interested in assaults which involved the brandishing of a sharp weapon but not injury and which were carried out on particularly vulnerable individuals. To provide access to sentences passed for cases with this sort of profile, all Scottish High Court convictions over a 5 year period were classified and recorded. The jurisdiction yielded a total of approximately 4500 cases over the period. Very many of the offences in the jurisdiction were charged under common law. A categorisation was designed which separated similar offences into discrete groups.

To cope with the variability in the characteristics which judges may take into account when sentencing, a study was carried out to establish the important features of each of the offence categories identified at the offence structuring stage. The intention of this knowledge engineering stage was to provide a basis for indexing each case to be entered in the system. Each identified attribute represents a single index. The SIS allows a user to select one or more indexes depending on whether they are perceived to be relevant to the conviction in hand. We characterise the use of a 'main' conviction together with a choice of characteristics as the principal offence approach to analysing previous similar cases. The similarity metric used to retrieve cases in the principal offence model is a direct match between the indexed characteristics of the offence and offender and the explicit interests of the judge in sentencing the case at hand.

A characteristic feature of CBS is that template instantiation and case retrieval define the types of cases which will be presented to the user for consideration. Similarities may be characterised as the quality of a case being such that its characteristics explicitly resemble those of the case at hand. For example, a judge could choose to retrieve all cases of assaults involving a sharp weapon, a vulnerable victim and no injury. Although such a definition of similarity may provide a good means of locating useful similar convictions, a database of 4500 cases will often provide only a handful of cases, and on many occasions, when several indexing attributes have to be set, no cases will be retrieved. In this approach, sentences for other cases which may also be regarded as analogous may also be missed

To overcome this limitation of the principal offence approach, a model was constructed which uses an alternative definition of similarity to group cases. The metric of similarity used in this model is a set of characteristics which can, for example, distinguish a seriously aggravated robbery from an aggravated robbery. These characteristics were identified by discussion with the judiciary as appropriate for the purposes of partitioning cases.

Convictions were re-classified using this technique and re-entered into the case base. The benefits of the approach based on composite similarities is that it greatly reduces the interaction required in order to produce a distribution of disposals related to convictions similar to the type under consideration. Instead of being required to make several distinct choices, the judge is invited to use a compound approach to a particular case which includes all relevant characteristics. The second benefit of this approach is that it results in distributions which contain larger numbers of relevant individual sentences. Such distributions provide a clearer impression of appropriate ranges of disposals.

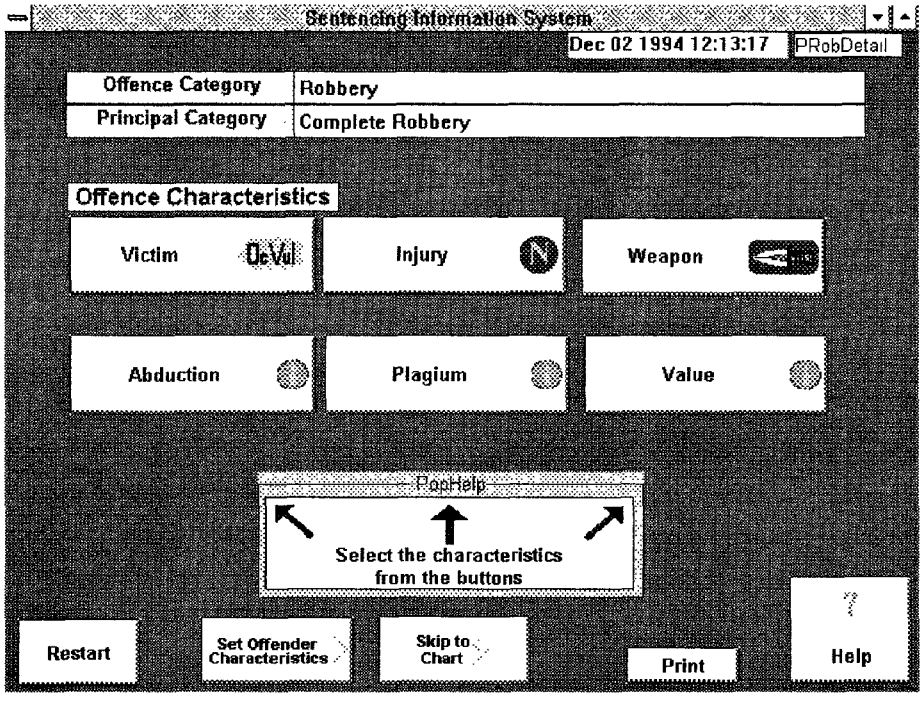

Figure 1: The offence characteristics page.

2.2 SIS Operation

The principal offence approach sub-system allows the user to select a particular offence category and the classification of the offence within that category. Having made these choices it is then possible to either select specific characteristics of the offence, select specific characteristics of the offender or view the data with neither offence nor offender characteristics specified. The latter course of action would be chosen to view the distribution of cases which are generally similar to the one at hand (i.e. the same classification) with a view to refining the data set by specifying further characteristics at a later stage. If the user elects to set the offence characteristics, s/he is presented with a list of the characteristics relevant to cases of the same type as the current case. The direct manipulation interface presents these characteristics as a set of buttons which display menus showing the possible settings of each characteristic. By default the current setting of a characteristic is shown by an icon on the button. All characteristics are initially set to 'Ignore': that is the characteristic will not be used to filter the cases selected (Figure 1).

When the characteristics have been set to the satisfaction of the user, it is possible to either display the distribution of sentences or to enter characteristics of the offender. The offender characteristics are entered in an analogous way to the offence characteristics through the use of buttons and menus.

The distribution of sentences is presented at several levels. At the top level the sentences are grouped on the basis of categories of disposal (Figure 2). Each category may then be further explored to show a distribution of the disposals of cases in that category. Further investigation of the distribution of each category reveals detail of the sentence imposed, together with reference information which would allow the user to retrieve case papers associated with individual disposals. At the lowest level, the system provides access to the summary of the details of the case which is held in the database (Figure 3).

Within the CBS paradigm, there is an acknowledged need for solutions with negative outcomes to be used to alert decision makers to potential problems concerning certain courses of action [16]. In the context of exploring sentencing possibilities, these negative outcomes can be equated with cases where a sentence was overturned on appeal. By the same argument, a sentence which was confirmed on appeal can be viewed as strongly supportive of a particular course of action. In order to provide support for the use of negative outcomes an appeal case-base can be explored from the point of view of the disposals imposed or supported by the Appeal Court. By default, data represented in distributions relates to first instance sentences. The user may explore whether a particular disposal was appealed and, if so, the nature of the outcome, by selecting particular groups of data and accessing the lowest level details about each particular case. As an alternative, data in displays may be limited only to those cases which went to appeal. The effect of this is to provide access to a very much restricted set of convictions, which nevertheless represent outcomes which are regarded as having greater significance in consideration of current courses of action by the sentencer.

2.3 SIS Implementation

The SIS was implemented using a proprietary interface-building tool [19]. This system provided a basis for eliciting user response in the form of menu choices. Menus are, in

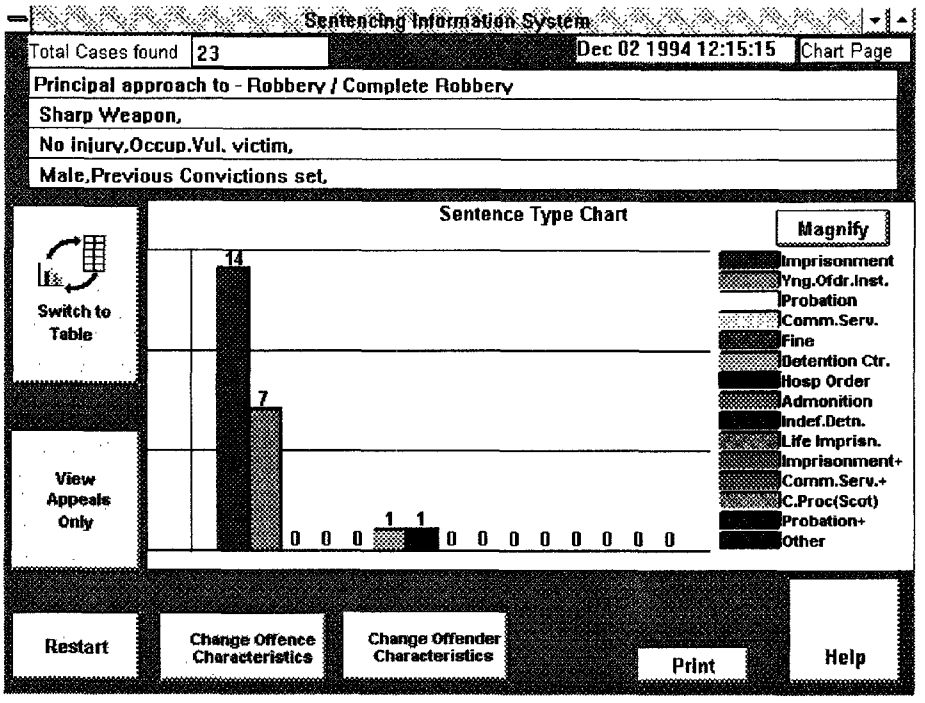

Figure 2: Initial chart showing disposals grouped by type.

general, hidden behind buttons to simplify the user interface. The system was designed to be presented to the user on a colour screen. While such software will run on any compatible hardware platform, the intention is that the system would be made available to the judiciary on a notebook machine capable of operating in stand-alone mode. The need for such a portable system is imposed by typical judicial work patterns.

The overall architecture allows the interface to collect the index choices from the user and to process these choices against the data store. The latter consists of data arranged in a series of relational database tables. Between the user and the case store are three sub-systems which control the retrieval of data from the case store and the characteristics which are offered to the user for selection. The sub-systems implement the principal offence approach, the modified offence approach and the appeals module. At all stages choices can only be made by input from the mouse, the keyboard being rendered inoperative to simplify user interaction. The choices made by the user are assembled into an SQL query such as that shown in Figure 4. This query is then used to retrieve the appropriate case from the database. Once data has been retrieved it can be explored by 'point and click'. Objects of interest such as bars on frequency distributions are sensitive to selection and able to reveal details about the cases referenced.

3 RESULTS

The process of involving a wide section of the target user community has been of considerable benefit to the project. The jurisdiction of Scotland is of manageable size and during the course of the development of the prototype nearly all of the judges were consulted. There was general agreement that the SIS was of significant utility in accessing relevant information, although a number of potential improvements were suggested, including a context-sensitive help system and improved navigational flexibility.

The direct manipulation interface was well received and judges found that they were rapidly able to construct meaningful queries and retrieve appropriate data despite having limited prior experience of computer use. Some users felt that this facility could be enhanced by using touch screen technology rather than mouse-driven input.

A preliminary evaluation which involved extended access to the system by four high court judges revealed that users were particularly interested in the facility for displaying the low-level details of cases which could be accessed by 'drill-down' from the distribution displays. This provided confirmation to the user that the cases extracted were of relevance to the conviction under consideration. There was, however, a reported need to support such summary case details with a more detailed description of the circumstances surrounding the sentence. This information is partly available in the narrative of the indictment but is also the basis of a judicial sentencing report which is completed for custodial sentences of 4 years or more. Text in the form of appeal court decisions was also identified as being of use to sentencers and could be incorporated in the lowest level of the system. Judges were particularly interested in the potential of the system for informing sentencing decisions on unusual cases.

The delivery of the SIS on a notebook personal computer was seen by users as a useful environment in which to work although concern was expressed about the speed of response of the initial prototype. A second prototype was produced which cut response time down to about 5 seconds. This was

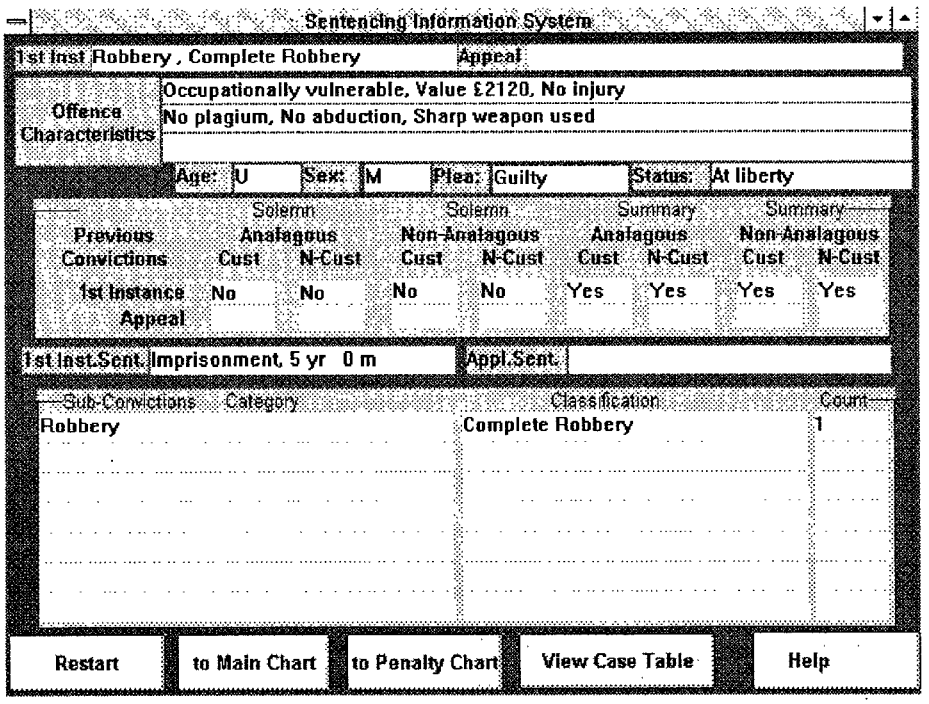

Figure 3: Low level conviction details.

seen as adequate but users showed evidence of habituation to this improvement. The advent of palm-top computers may provide for less intrusive delivery of the SIS but further research to improve performance is important if applications of this sort are to be widely accepted.

4 CONCLUSIONS

The SIS system was constructed as a prototype to enable exploration of the concept of a computer-based tool for assisting sentencers to retrieve relevant cases. A particular problem with the domain is that given a limited number of available cases it is important to balance refinement of case selection with the need to retrieve enough similar cases to provide a satisfactory distribution of disposals. Further refinement of the offence classification is required to tune the system so that data retrieved is of maximum benefit to end users. A significant problem is presented by the limitations of performance which are inherent in the combination of resource-hungry applications and current personal computer hardware capabilities. One outcome of informal observations of the use of the system by members of the judiciary is that there is a tendency to use the system in a manner similar to a spreadsheet to provide a 'what-if' facility. The implications of adding in offence and offender characteristics can be explored cumulatively by adding in each characteristic singly and immediately observing the effect on the distribution of disposals. The prototype, however, produces displays using SQL grouping queries which are relatively time-consuming for the data server to process, thus limiting the use of the tool for exploratory purposes. The potential benefits of dynamic query systems have been identified in connection with applications of this type [20]. In

```
select sentence_type, count(*)
     from robberies
     where offence_category = 'complete robbery'
     and victim = 'occupationally vulnerable'
     and injury = 'none'
     group by sentence_type;
```

Figure 4: SQL query generated by choices in Figure 1

this approach, the display is continually updated to match the setting of selection objects (buttons or sliders) giving users immediate feedback of the impact of their choices. This technology is only marginally feasible using currently available tools. We are developing research interests in high performance database systems using data compression as a means of delivering data to applications such as the SIS at a rate sufficient to satisfy observed needs of users.

5 ACKNOWLEDGEMENT

This research was greatly assisted by the Lord Justice-Clerk, The Rt. Hon. Lord Ross and funded by The Scottish Office, Home and Health Department. Programming work was carried out by Mr. J.D. Wilson.

References

[1] Grover, V., K.Fielder, and J.Teng, Exploring the success of Information technology enabled business process reengineering. IEEE Transactions on Engineering Management. 41: 3 (1994), 276-284.

[2] Wilson, F., and J.Wilson, The role of computer systems in organizational decision making. The Information Society. 10: 3 (1994), 173-180.

[3] Chan, J. A computerised sentencing information system for New South Wales courts. Computer Law and Practice. (1991), 137-150.

[4] Doob, A., and N.Park, Computerized sentencing information for judges: an aid to the sentencing process. Criminal Law Quarterly. 30: 1 (1987), 54-72.

[5] Murbach, R., and E.Nonn, Sentencing by artificial intelligence tools: some possibilities and limitations. Joint Meeting of the Law and Society Association and the International Sociological Association. Amsterdam. (1991).

[6] Murbach, R., and E.Nonn, Similarity in harder cases. Proceedings of the 4th International Conference on Artificial Intelligence and Law (Amsterdam, June, 1993), 236-244.

[7] Simon, E., G.Gaes, ASSYST - Computer support for guideline sentencing. Proceedings of the 2nd International Conference on Artificial Intelligence and Law (Vancouver, June, 1989), 195-200.

[8] Bainbridge, D. CASE: Computer assisted sentencing in magistrates' courts, Proceedings of BILETA Conference, 1991.

[9] Bain, W. Judge. In Inside case-based reasoning, Reisbeck and Schank (eds), Erlbaum, Northvale, NJ, 1989.

[10] Kolodner, J. An introduction to case-based reasoning. Artificial Intelligence Review. 6: 3-34 (1992), 3-34.

[11] Kolodner, J. Case-based reasoning. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[12] Simoudis, E. Using case-based retrieval for customer technical support. IEEE Expert. (1992) 7-12.

[13] Skalak, D., and E.Rissland, Arguments and cases: an inevitable intertwining. Artificial Intelligence and Law. 1: 1 (1992), 3-44.

[14] Sanders, K. Representing and reasoning about open-textured predicates. Proceedings of the 3rd International Conference on Artificial Intelligence and Law (Oxford, June, 1991), 137-144.

[15] Hansen, J., R.Meservy and L.Wood, Indexing tree and pruning concepts to support case-based reasoning. Omega. 22:4 (1994),361-369

[16] Kolodner, J. Improving human decision making through case-based decision aiding. AI Magazine. (Summer 1991), 52-68.

[17] Shneiderman, B. Direct manipulation: a step beyond programming languages. IEEE Computer. 16:(1983), 57-69.

[18] Frohlich, D. The history and future of direct manipulation. Behaviour and Information Technology. 12: 6 (1993), 315-329.

[19] Gentium 1.7.1, Planning Sciences Ltd, Ipswich.

[20] Shneiderman, B. Dynamic queries for visual information seeking. IEEE Software. (1994), 70-77.