SYSTEM EVALUATION OF A KNOWLEDGE-BASED DATABASE DESIGN TOOL: AN APPLICATION OF THE THESAURUS APPROACH

Shahrul Azman Mohd Noah
Faculty of Science and Information Technology
Universiti Kebangsaan Malaysia
43600 UKM Bangi
Selangor, Malaysia
Tel.: 603-8292482
Fax: 603-8255732
email: samn@sun1.ftsm.ukm.my

Michael Williams
Department of Computing and Mathematics
Manchester Metropolitan University,
John Dalton Building, Chester Street,
Manchester M1 5GD, UK
Tel.: 0161-247-1500
Fax: 0161-247-1483
email: m.williams@doc.mmu.ac.uk

ABSTRACT

The use and exploitation of real-world knowledge by a knowledge-based (KB) database design tool was claimed capable of enhancing its performance in terms of processing efficiency, quality of designs produced and appearance of tool's intelligence. However, as to date, there have been no serious attempts made to evaluate these claims. This paper presents such an evaluation of one approach proposed to facilitate the system storage and exploitation of real-world knowledge; the thesaurus approach. Results obtained have demonstrated that with-in the scope of design synthesis, the claim for an increase in processing efficiency has been achieved. However, within the context of design diagnosis all the aforementioned claims have been met by the thesaurus approach.

Keywords: ***Database design tool, Knowledge-based database design tool, Thesaurus approach, Evaluation***

1.0 INTRODUCTION

Recent years have seen the development of a number of knowledge-based (KB) database design tools that employ expert system technology in order to provide support to a human designer during the process of database analysis and design [1, 2, 3]. Such tools are generally intended to act as assistants to human designers [4], being capable of providing guidance, proposing alternative solutions, and investigating the consequences of design decisions [5].

The effectiveness of existing tools has demonstrated the viability of representing database design expertise in a computer program, however, observing such systems in use makes it clear that human designers contribute far more than database design expertise to the design process [6]. Human designers, even when working in an unfamiliar domain, are able to make use of their knowledge of the real world in order to interact with users, make helpful suggestions and inferences, and identify potential errors and inconsistencies [7, 8]. Conversely, the majority of existing KB database design tools do not possess such real-world knowledge, and are therefore required to ask many questions during a design session that may be viewed as being trivial [9, 10]. This situation has resulted in numerous calls for the representation of real-world knowledge within such tools, coupled with the ability to reason with and make use of this knowledge.

A number of approaches to representing and exploiting such real-world knowledge have been proposed, including the *dictionary* approach [11], the *thesaurus approach* [5, 9], and the *knowledge reconciliation approach* [6, 10]. These approaches have been accompanied by various claims [7, 9] that the use of such knowledge has the potential to increase the appearance of tools' intelligence, to improve the quality of the designs produced, and to increase processing efficiency. However to date, little if any formal evaluation of these claims has taken place. This paper presents an evaluation of the thesaurus approach [5, 9] as originally employed by the Object Design Assistant (ODA) [1, 12], the intention being to initiate the gathering of evidence to support or otherwise the claims previously stated. The thesaurus approach was chosen mainly because it has been used to assist users in designing object-oriented databases. Furthermore, the evaluation of the dictionary approach and the knowledge reconciliation approach have been previously described in [13, 14].

2.0 THE INTELLIGENT OBJECT ANALYSER

In order to explore and evaluate the capabilities and benefits gained from exploiting this approach, a prototype KB database design tool, the *Intelligent Object Analyser* (IOA), was developed. IOA provides support for the design of the structural (data) aspects of object-oriented databases.

The intended user is a database designer or a systems analyst, who is familiar with systems modelling concepts and the domain to be modelled. Knowledge of object-oriented databases or of object-oriented analysis and design techniques is not a requirement. It is not the purpose of this paper to discuss IOA in depth, however, a brief outline of the structure and method of operation is required in order to illustrate how the real-world knowledge may be represented and exploited during design processing. An overview of the structure and activities performed by the IOA therefore follows.

The current version of the IOA tool runs in a PC environment, and was developed using Common LISP (Allegro CL\PC). The IOA knowledge base contains a mixture of rules and facts. Rules correspond to knowledge of how to perform the design task (the order in which design activities take place), detecting and resolving ambiguities, redundancies and inconsistencies within an evolving design, and handling the gradual augmentation of an evolving design as a design session progresses. Facts are used to represent two views of the application domain; an initial representation (the problem domain model) as provided by the user, and the object-oriented design generated from this initial representation.

During a design session, IOA adopts a two-step procedure, which is as follows:

- The first step involves creating an initial representation of the application domain (known as the problem domain model) and the subsequent refinement of this model
- The second step involves the refinement of the problem domain model by detecting and resolving any inconsistencies that may exist, and the transformation of the model into object-oriented form (also known as the analysis model).

The first stage of processing requires a set of declarative statements that describe the application domain to be submitted to IOA. These statements are a variation of the method of interactive schema specification described by Baldiserra *et al*. [15] being based upon the binary model described by Bracchi *et al*. [16]. Each statement links together two concepts (taking the form *A verb-phrase B*), and falls into one of three classes of construct, corresponding directly to the structural abstractions of association, generalisation, and aggregation. The statements are used to construct a problem domain model representing the application domain. Once constructed, IOA attempts to confirm its "understanding" of the semantic aspects of the problem domain model; that is, whether each structure within the model represents generalisation, aggregation or association.

The problem domain model upon construction is submitted to a series of refinement procedures to detect and resolve any inconsistencies (such as redundancies that may be present within generalisation hierarchies) that may exist. These procedures are performed both with and without the requirement of user input (sometimes referred to as *external* and *internal* validation respectively). Once such inconsistencies have been identified and resolved, IOA makes use of the problem domain model in order to generate a conceptual model (in object-oriented form).

The detail design sequence for the aforementioned two-step procedure is conceptually illustrated in Fig. 1.

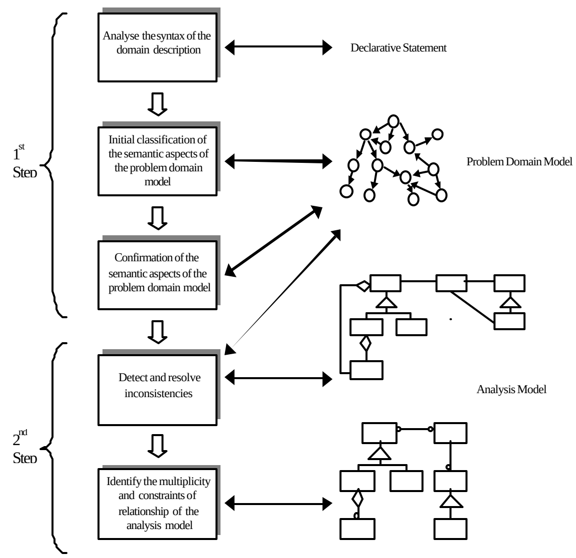

Fig. 1: The detail design sequences of the IOA two-step procedure

As previously discussed, the IOA tool has been developed in order to assist with a series of experiments aimed to evaluate the contribution of real-world knowledge to the activities of KB database design tools. To facilitate this aim, IOA is capable of conducting design sessions both with and without making use of real-world knowledge.

3.0 REPRESENTING THE THESAURUS APPROACH

The thesaurus approach to representing and exploiting real-world knowledge by a KB database design tool is illustrated by the Object Design Assistant (ODA) tool of Lloyd-Williams [5, 9]. The approach represents real-world knowledge as generic models comprising of domain specific concepts, linked together via abstraction mechanisms of aggregation, generalisation and association. The domain concepts may be referred to by any number of associated synonyms where appropriate. The similar representation of real-world knowledge has been implemented in the IOA tool.

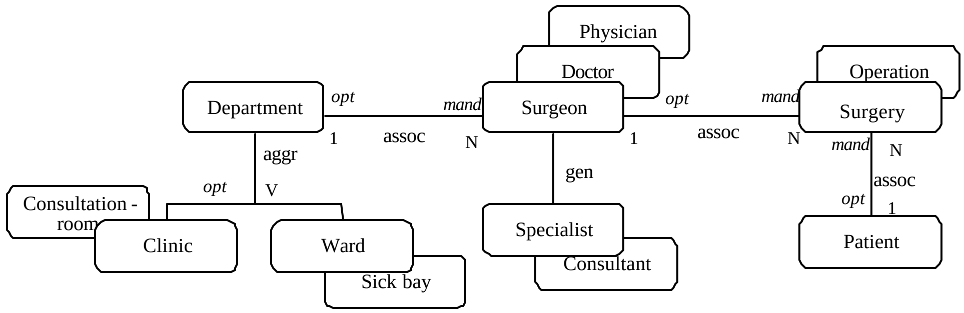

Fig. 2: Fragment of real-world knowledge for a health services domain represented using the thesaurus approach

Fig. 2 illustrates an example of real-world knowledge represented using the thesaurus approach.

As can be seen in Fig. 2, the thesaurus-type structure exhibits flexibility, allowing each concept to be referred to by any number of associated synonyms, and the abstraction mechanisms linking concepts to take any name provided by the user. For example, the following statements would all be recognised as semantically equivalent by the IOA (with reference to Fig. 2) concerning a single association relationship between two concepts.

SURGEON PERFORMS OPERATION
SURGERY PERFORMED-BY SURGEON
DOCTOR CONDUCTS OPERATION
OPERATION DONE-BY PHYSICIAN

The content of this structure can also be classified as "information rich" as it contains information about integrity constraints as well as memberships participation (whether optional or mandatory) for links between pairs of concepts.

The IOA tool is capable of processing in two different modes, without the use of real-world knowledge (basic mode) and using real-world knowledge provided by the thesaurus approach (thesaurus mode). The basic mode of processing has been previously described in section 2 of this paper. At various stages during the basic mode of processing, the IOA conducts a dialogue with the user to confirm its understanding of the application domain or to obtain additional information. When making use of real-world knowledge provided by the thesaurus approach, the tool refers to this knowledge wherever possible. The tool resorts to questioning the user; only when the real-world knowledge cannot provide the required information.

This section has provided only a brief overview of the method of knowledge representation employed by the thesaurus approach. Those interested with further details of this approach, along with the claimed benefits associated with it use, are referred to the relevant source literature (see for instance [5, 9]).

4.0 EMPIRICAL TESTING AND EVALUATION STRATEGIES

KB systems (KBSs) or expert systems can be evaluated in terms of component evaluation or system evaluation [17]. Component evaluation is concerned with the examination and assessment of the individual components of a KBS, such as rules, weights or frames, whereas system evaluation is concerned with the examination and assessment of the performance of the system as a whole. Given this description, the empirical testing and evaluation process conducted in this study falls into the category of system evaluation.

In system evaluation, the case-testing method was declared by O'Keefe and Preece [18] as the most dominant and prevalent method. This method usually involves the execution of a set of test-cases by a KBS where the results produced are compared with those of human experts or computational models (such as simulation models or regression models). This method has exhibited satisfactory results, and is likely to be a popular approach in the future [19]. The case-testing method was, therefore, seen suitable to be employed in this study.

In adopting the case-testing method, this study adopts the following steps; each of these is elaborated in the next section.

1. Establish a set of performance-related criteria based on which a KB database design tool can be evaluated.
2. Generate a representative set of test-cases to provide adequate coverage for each type of tests. Execute the test cases on the tool with and without the use of real-world knowledge.
3. Results obtained from executing the test cases using the real-world knowledge are compared with the results obtained when no such knowledge is in use.

4.1 Establishing Criteria

The establishment of criteria of which a KB database design tool can be evaluated was found to be central in the testing and evaluation activities. The set of criteria established within the scope of this research work focuses on the aspects of system's processing efficiency, quality of a design model and appearance of system's intelligence.

The main criteria of interest used are as follows:

- *Processing time*. Processing time refers to the CPU time required by a tool to perform a single design action. It is not influenced by human factors as it is measured from the point at which the tool commences an action until that action is complete. Processing time is, however, influenced by the complexity of the design input; the complexity of the system-held domain knowledge and the reasoning associated with it; and the specification of the processor of the personal computer in use.
- *User/tool interaction*. User/tool interaction refers to the number of interactions required between the tool and the user, in order for the tool to confirm its understanding of some aspects of the application domain or to acquire additional information should it be required.
- *Completeness of the designs produced*. Completeness is defined as the ability of a data model to meet all the user information requirements [20]. Within the scope of the testing performed, completeness is measured in terms of the number of missing classes and relationships associated with the design example used.
- *Detected and resolved errors*. This criterion refers to the number of errors (previously synthesised and used to generate a range of test-cases) that have been successfully detected and resolved by the tool. The errors included synonymous class(es), synonymous relationship(s) and combination of both.
- *Errors within designs produced*. This criterion directly relates to the *number of errors resolved* criterion. The synthesised errors that remain undetected by the tool are continuously processed by the tool and generated as part of the design produced. The errors within designs produced criterion is, therefore, measured in terms of the number of these undetected synthesised errors.
- *Suggestion of missing design elements*. This criterion measures whether the elements (within the generated design) are based entirely upon user-provided information, or are included as a direct result of the system consulting with its real-world knowledge.

The above criteria were used to assess the IOA performance, in terms of processing efficiency, quality of designs produced and appearance of intelligence.

The increase in processing efficiency can be judged from the reduction in the number of user/tool interactions and processing time required by a tool [5, 21]. Fewer number of user/tool interactions required may indicate some aspects of the confirmation and augmentation process are internally confirmed and processed; and decreased processing time may indicate that some aspects of design processing are no longer required or are simplified.

The quality of designs produced can be judged from their completeness and consistency. Therefore, the criteria *completeness of designs produced* and *errors within design produced* can be used to assess this performance factor.

The justification of the appearance of tool intelligence can be judged from the extent to which a tool is capable of simulating the approach taken by a human designer when performing a design task, such as offering suggestions and inferences, identifying potential errors and inconsistencies, and forwarding questions to the user only when necessary [6, 7]. The elements such as the number *of user/tool interactions*, the number of *errors capable* of being *detected and resolved* by the tool, and *suggestion of missing design elements* were, therefore, seen as suitable for evaluating this claim.

Fig. 3 summarises the association between each criteria to the performance factors (i.e. which criteria are used to assess which performance factors).

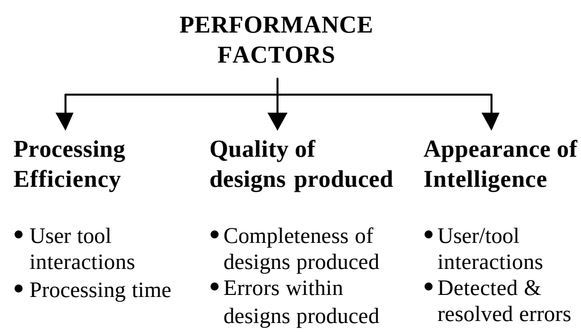

Fig. 3: Summary of the tested criteria used to assess the performance of a KB database design tool

4.2 Generation and Execution of Test-Cases

The set of test-cases used during the testing were generated from a set of design problems which were primarily extracted from the available literature. The advantage of this method is that the accompanying solution could be used as a benchmark and compared with the IOA-suggested solution in order to confirm the appropriateness or otherwise of the designs produced [13]. The synthesisation of multiple test-cases from existing design problem instead of using the real-world test-cases which are not easily available has been previously proposed by O'Keefe and O'Leary [17] and implemented in [19] and [22].

The approach to design processing employed by KB database design tool was categorised as *design synthesis* and *design diagnosis* [23]. Design synthesis is where the tool has the capability of generating design output, whereas, design diagnosis is where the tool detects any inconsistencies or redundancies and suggests corrections in design. In order to evaluate the performance of the IOA tool within both approaches, two types of tests have been implemented. The first test (Test A) involved the generation and execution of a set of test-cases with varying degrees of complexity, and the second test (Test B) involved the generation and execution of a set of test-cases with a different number and combination of types of errors. The intention of Test A and Test B is to assess as whether the information and reasoning associated with the use of the thesaurus approach capable of enhancing the performance (processing efficiency, quality of designs produced and appearance of tool's intelligence) of the IOA tool within the scope design synthesis and design diagnosis respectively.

In performing Test A, each of the examples of design problems was systematically altered by dividing them into multiple test-cases with varying degrees of complexities [13]. Within the scope of the testing, the complexity of a design test-case is defined as the number of concepts, and the relationships between the concepts [24, 25].

In the second test (design diagnosis), each of the example design problems was systematically embedded with a series of intentionally-synthesised errors to generate a number of test-cases [14]. Each of these generated test-cases contains a combination of different types and numbers of synthesised errors, including synonymous concept(s), synonymous relationship(s) and combinations of both.

As previously discussed, the IOA is capable of processing in two different modes: processing without the use of real-world knowledge (basic mode) and processing with the use of real-world knowledge represented as the thesaurus approach (thesaurus mode). The test-cases generated were then executed in both modes of processing. Thus, for each test-case, two sets of results were obtained and compared, following an approach recommended by O'Keefe and Preece [18].

4.3 Method of Analysis of Results

The quantitative method of analysis was employed because all the selected criteria are quantifiable. In this study, the results observed from the execution of test-cases in thesaurus mode were subsequently compared with the results observed from the execution of test-cases in basic mode. A statistical *hypothesis test* was then conducted to find if there exist any significant differences between these observed results at the 5% of significance level.

Although there are several recommended statistical methods available to test such hypotheses, the paired **t**-test method is highly appropriate in such circumstance as those prevailing in this study [17, 26]. The paired **t**-test method is a form of *repeated measures design*, where the same variable (observed criterion) is measured on several occasions (processing modes) for each subject (test-case).

5.0 RESULTS AND FINDINGS

The results presented here emanate from a series of tests performed on health services domain problem found in the general literature. A total of 26 and 35 test-cases were generated for Test A and Test B respectively. A discussion of the analysis of the observed results for each type of test follows.

5.1 Approach to Design Synthesis (Test A)

The objective of Test A was to validate the hypothesis associated with the use of real-world knowledge represented as the thesaurus approach by IOA within the scope design synthesis. Therefore, the set of null hypotheses set up for this test is as follows (corresponding to each of the criteria under observation in this test):

User/Tool Interactions	H_0 – Exploiting the thesaurus approach does not reduce the number of user/tool interactions required per complexity.
CPU time	H_0 – Exploiting the thesaurus approach does not reduce the CPU time required per complexity.
Suggested elements	H_0 – Exploiting the thesaurus approach does not increase the number of suggested elements per test.
Completeness	H_0 – Exploiting the thesaurus approach does not increase the completeness of the designs produced per test.

Based upon the paired **t**-test results presented in Table 1 it is apparent that there are significant differences between the thesaurus approach and the basic approach in terms of the number of user/tool interactions and the CPU time required per complexity. The observed significance level and the negative **t**-Value suggest that the null hypothesis should be rejected for both criteria. It may, therefore, be stated that the thesaurus approach increased the overall processing efficiency by reducing the number of user/tool interactions and the CPU time required per complexity.

Table 1: Paired **t**-test results - the thesaurus and basic approaches (Test A)

Criteria	df	t-Value	Sig. t (P)
User/tool interaction per complexity	25	-4.52	0.000
CPU time per complexity (sec.)	25	-2.91	0.010
Suggested elements per test	25	N/A	N/A
Completeness per test	25	N/A	N/A

However, Table 1 also indicates that the aspects of completeness and the suggested elements per test were not significantly different between the two approaches (the statistical test was invalid as the thesaurus and the basic approaches do not provide suggestions for missing information, therefore, both approaches result in similar numbers of missing elements within the resulting designs). Accordingly, the use of thesaurus approach has not resulted in an improvement in the quality of the resulting design output (measured in terms of increasing the completeness of the designs produced).

The significant reduction in the number of user/tool interactions required suggests an increase in the appearance of intelligence of the tool. However, the (statistical) non-significance of the suggested elements per test criterion may be viewed as jeopardising this claim.

5.2 Approach to Design Diagnosis (Test B)

The objective of Test B was to validate the hypothesis associated with the use of real-world knowledge represented as the thesaurus approach within the scope of design diagnosis.

The hypothesis of this test was that the use of real-world knowledge represented as the thesaurus approach increases the tool's processing efficiency (by reducing the processing time and number of user/tool interactions required to detect and remove errors), increases the quality of designs produced (by minimising the number of errors per design produced) and increases the appearance of tool's intelligence (by increasing the number of errors detected and resolved and by reducing the number of user/tool interactions required to solve errors). As a result, the set of null hypotheses set up are as follows:

User/Tool Interactions	H_0 – Exploiting the thesaurus approach does not reduce the number of user/tool interactions required per error detected and resolved.
CPU time	H_0 – Exploiting the thesaurus approach does not reduce the CPU time required per error detected and resolved.
Errors per design output	H_0 – Exploiting the thesaurus approach does not reduce the number of errors within designs produced per test.
Errors resolved per test	H_0 – Exploiting the thesaurus approach does not increase the number of errors detected and resolved per test.

As can be seen in Table 2, the number of user/tool interactions and CPU time required per error were significantly reduced by the thesaurus approach as evidenced from the significant value of **P** (**P** < 0.05) and the negative **t**-Values. It may therefore be claimed that the thesaurus approach has the capacity to increase the overall processing efficiency of the IOA tool by reducing the number of user/tool interactions and the CPU time required per error.

Table 2: Paired **t**-test results - the thesaurus and basic approaches (Test B)

Criteria	df	t-Value	Sig. t (P)
User/tool interaction per error	34	-7.83	0.000
CPU time per error (sec.)	34	-3.83	0.001
No. of errors per design output	34	-6.24	0.000
No. of errors resolved per test	34	6.24	0.000

Table 2 also illustrates that the number of errors per design produced was significantly reduced when using the thesaurus approach. Accordingly, the claim for an improvement in the quality of designs produced (minimised the number of errors within designs produced) has therefore been met by the IOA tool when using thesaurus approach.

The significant increase in the number of errors detected and resolved per test initially indicates an increase in the appearance of tool intelligence. This indication was further supported by the significant reduction in the number of user/tool interactions required per error. As a result, the claim for an overall increase in the appearance of intelligence of the IOA tool has been achieved when using the thesaurus approach in terms of these criteria.

6.0 DISCUSSION AND CONCLUSIONS

Table 3 presents a summary of the conclusions reached for both tests.

Table 3: Summary of conclusions

Performance Factors	Approach to Design	
	Design Synthesis	Design Diagnosis
Increases overall tool processing efficiency	Yes	Yes
Improves quality designs produced	No	Yes
Increases overall appearance of tool intelligence	Unjustifiable	Yes

Within the tool approach to design synthesis, only the claim for an overall increase in processing efficiency has been met by the thesaurus approach. The conclusion ensued as a result of the significant reduction in the number of user/tool interactions and CPU time required for each increase in complexity. These reductions occurred to a certain extent due to the "information rich" contents held by the thesaurus approach. For instance, constraints and membership requirements related information have the potential to impact upon the performance-related criteria.

Within the context of design diagnosis, exploiting the thesaurus-type structure during processing has managed to simplify the process of detecting and resolving errors (i.e. certain aspects of confirmation and processing were no longer required). This was evidenced from the significant reduction of the CPU time and user/tool interactions required, which indicates that the objective of increasing the overall processing efficiency (with the use of the thesaurus approach) has been met. Furthermore, using this approach in design diagnosis has also increased the tool's appearance of intelligence. The significant reduction in the number of user/tool interactions required to detect and resolve errors and the increased in the number of errors detected and resolved have supported this conclusion.

Although the objective of improving the quality of designs produced when using the thesaurus approach was not fulfilled during the evaluation of the tool approach to design synthesis, it was fulfilled during the evaluation of the tool approach to design diagnosis. This conclusion was reached as a more consistent design output has been produced (the number of errors per design produced has been reduced).

Very encouraging results have been obtained from the evaluation work, nevertheless, it is recognised that consideration must be given to a number of practical issues. The effectiveness of the tool depends greatly on the accuracy and completeness of the system-held real-world knowledge, and the results obtained from the tests may be influenced to a certain extent by the variety and coverage of the generated test-cases.

7.0 SUMMARY AND FUTURE WORK

This paper has presented the findings of an assessment of the thesaurus approach to representing and exploiting real-world knowledge by an intelligent database design tool; the IOA. The use of encapsulated real-world knowledge by the IOA tool when performing the tasks of design synthesis has yielded the benefit in terms of increasing processing efficiency. On the other hand, when performing the tasks of design diagnosis, the exploitation of the thesaurus approach has yielded the benefits of increasing processing efficiency, improving the quality of designs produced and increasing the appearance of tool's intelligence.

Ongoing work includes extending the series of evaluative experiments to include the assessment of other approaches recommended for representing and exploiting real-world knowledge by KB database design tools.

REFERENCES

[1] M. Lloyd-Williams, et al, "Design and Development of an Intelligent Database Design Tool" in *IC-AI '99: Proceedings of the 1999 International Conference on Artificial Intelligence*, 1999, Las Vegas, USA.

[2] S. A. Noah "Knowledge-Based CASE Tool for Object-Oriented Database Design", in: *Malaysian Science and Technology Congress* 1999, Johor Bahru, Johor.

[3] Storey et al, "Database Design with Common Sense Business Reasoning and Learning", ACM Transactions on Database Systems, Vol. 22, No. 4, 1997, pp. 471-512.

[4] I. Vessey and A. P. Sravanapudi, "CASE Tools as Collaborative Support Technologies". *Communications of the ACM*, Vol. 37, No. 1, 1995, pp. 83-102.

[5] M. Lloyd-Williams, "Exploiting Domain Knowledge During the Automated Design of Object-Oriented Databases", in *Proceedings of the 16th International Conference on Conceptual Modelling, California*. Spinger-Verlag, 1997, pp. 16-29.

[6] V. C. Storey et al, "A Common-Sense Reasoning Facility Based on the Entity-Relationship Model", in: *Proceedings of the 12th International Conference on the Entity Relationship Approach, Berlin*, Springer-Verlag, 1993, pp. 218-229.

[7] V. C. Storey and R. C. Goldstein, "Knowledge-Based Approach to Database Design". *Management Information Systems Quarterly*, Vol. 17, No. 1, 1993, pp. 25-46.

[8] V. C. Storey, "Real World Knowledge for Databases". *Journal of Database Administration*, Vol. 3, No. 1, 1992, pp. 1-19.

[9] M. Lloyd-Williams, "Knowledge-Based CASE Tools: Improving Performance Using Domain Specific Knowledge". *Software Engineering Journal*, Vol. 9, No. 4, 1994, pp. 167-173.

[10] V. C. Storey et al, "Knowledge Reconciliation for Common Sense Reasoning", in: *Proceeding of the 4th Annual Workshop on Information Technologies and Systems, Univ. British Columbia, Vancouver*, 1994, pp. 87-96.

[11] A. Kawaguchi et al, "An Intelligent Interview System for Conceptual Design of Database", in: *ECAI '86: The 7th European Conference on Artificial Intelligence, Stockholm*, Conference Services Ltd., 1986, pp. 1-7.

[12] M. Lloyd-Williams, "Expert System Support for Object-Oriented Database Design". *International Journal of Applied Expert Systems*, Vol. 1, No. 3, 1993, pp. 197-212.

[13] S. A. Noah and M. Lloyd-Williams, "An Evaluation of Two Approaches to Exploiting Real-World Knowledge by Intelligent Database Design Tools", in: *Proceedings of the 17th International Conference on Conceptual Modelling, Singapore*, Springer-Verlag, 1998, pp. 197-210.

[14] S. A. Noah and M. Lloyd-Williams, "Exploitation of Domain Knowledge by Knowledge-Based Database Design Tools: the Dictionary Approach", in: *Proceedings of the 3rd Joint Conference on Knowledge-Based Software Engineering, Smolenice, Slovakia*. Amsterdam: IOS Press, 1998, pp. 250-257.

[15] C. Baldiserra et al, "Interactive Specification and Formal Verification of User's Views in Database Design", in: *Proceedings of the 5th International Conference on Very Large Databases, Rio de Janeiro, Brazil*, 1979, pp. 262-272.

[16] G. Bracchi et al., "Binary Logical Associations in Data Modeling". in: Nijsen, G. M. (ed.) *Modeling in Data Base Management Systems*. Amsterdam: North-Holland, 1976, pp. 125-148.

[17] R. M. O'Keefe and D. E. O'Leary, "Expert System Verification and Validation: A Survey and Tutorial", *Artificial Intelligence Review*, Vol. 7, No. 1, 1993, pp. 3-40.

[18] R. M. O'Keefe and A. D. Preece, "The Development, Validation and Implementation of Knowledge-Based Systems", *European Journal of Operational Research*, Vol. 92, No. 3, 1996, pp. 458-473.

[19] A. J. Gonzalez et al, "Performance Evaluation of a Large Diagnostic Expert System Using a Heuristic Test Case Generator", *Engineering Application of Artificial Intelligence*, Vol. 9, No. 3, 1996, pp. 275-284.

[20] D. L. Moody, Metrics for Evaluating the Quality of Entity Relationship Model, in: *Proceedings of the 17th International Conference on Conceptual Modelling, Singapore*, Springer-Verlag, 1998, 211-225.

[21] C. Y. Suen et al, "Verifying, Validating, and Measuring the Performance of Expert Systems", *Expert Systems With Applications*, Vol. 1, 1990, pp. 93-102.

[22] L. A. Miller, "Dynamic Testing of Knowledge-Bases Using the Heuristic Testing Approach", *Expert Systems with Applications*, Vol. 1, No. 3, 1990, pp. 249-269.

[23] R. Oxman, R. and J. S. Gero, "Using an Expert System for Design Diagnosis and Synthesis", *Expert Systems: The International Journal of Knowledge Engineering*, Vol. 4, No. 1, 1987, pp. 4-14.

[24] S. Kesh, "Evaluating The Quality of Entity Relationship Models", *Information and Software Technology*, Vol. 37, No. 12, 1995, pp. 681-689.

[25] N. Pippenger, "Complexity Theory", *Scientific American*, Vol. 238, No. 6, 1978, pp. 90-102.

[26] R. M. O'Keefe et al, "Validating Expert System Performance", *IEEE Expert*, Vol. 2, No. 4, 1987, pp. 81-90.

BIOGRAPHY

Shahrul Azman Mohd. Noah received his Master and Ph.D. from the University of Sheffield in 1995 and 1998 respectively. Currently, he is a lecturer at the Department of Information Science, Faculty of Information Science and Technology, National University of Malaysia. His research interests include databases, automated information systems engineering and intelligent information agent.

Michael Williams is currently the Professor of Computing and Head of the Department of Computing and Mathematics at the Manchester Metropolitan University. His research interests include automated systems/software engineering, the knowledge discovery process (KDD), data mining/warehousing, information systems evaluation, information systems failure, information requirements, information system in healthcare. He has published a number of papers related to this area.