Research Report pjhk/ap,11/2000

Enhancing database technology to better manage and exploit Partially Structured Data

Prof P J H King and Dr A Poulovassilis

1 Introduction

An important and growing theme in software technology is the need for better methods for storing and managing large volumes of information in ways that enable the full potential of its content to be exploited in applications. Current mainstream database products, both relational and object-based, target applications that conform in the main to the *type/instance* paradigm whereby a number of types are defined as metadata (the schema) and the database consists of collections of instances conforming to these types; database update mainly comprising insertion, deletion and modification of these instances.

For many important application areas this paradigm is appropriate and effective. However, there are other important application areas, of which *crime*, *fraud* and *road-accident investigation* are good examples with which the Group has some familiarity, for which the paradigm is not adequate. In these application domains the original information comes in two forms: some structured data which conforms to the current concept of a database schema, and some free text which may contain domain-specific terminology and/or be stylised in a domain-specific way. Whilst some database products do now provide for the inclusion of quite large free-text objects if specified as such in the schema, there is little effective provision for integrating their processing.

A particular example of the type of information we have in mind occurs in the normal Police response to a burglary, which will include the initiation of some form of computer record. Present systems will have provision for information to be formatted against a database schema designed to cover basic information such as incident class, time, date, location, name of person reporting, victim's name, number of the officer taking the call, and so on, with appropriate screen formats to facilitate their entry. However, it is not to be expected that all relevant information can be provided for in this way. Thus, there must also be provision for the entry of significant amounts of free text which will contain further important, and often *the* most important, information. This text is likely subsequently to be the subject of various forms of keyword search ("Do we have any reference in our burglary records to valuable antique bronze flamingoes?") and also be directly displayed or printed for human reading, but it will not otherwise be processed. In our view it should be and there is a need to investigate how.

An extreme reaction to the limitations of current database products has been the call to abandon the concept of metadata encoded in a schema and for there to be research into the management of extensive unstructured collections of largely free-text information, sometimes termed *text-mining*. We believe that such an approach is unrealistic for the broad class of applications that we have in mind. Moreover it ignores the fact that information can often be more effectively represented in ways other than as free text [46, 47]. In our view, where information can be described by a schema then it should be, since advantage can then be taken of effective and efficient processing using well developed methods.

In the next section of this report we define partially stuctured data and briefly indicate the kind of facilities future DBMS should provide for its successful exploitation and management. In section 3 we outline our plans for progressing the research, for developing a methodology upon which the future DBMS we envisage could be based, and for producing appropriate experimental software to evaluate the ideas in a practical context. In section 4 we argue the importance of this work in the broader context of the development of future DBMS, their importance, and the commercial possibilities. In section 5 we give in some detail previous research which we believe should be studied and evaluated at the outset of the research and of work in computational linguistics and natural language understanding which we might use. Section 6 briefly reiterates and summarises our views and the opportunites which this work presents.

2 Partially Structured Data

Information or data which is partly formatted in accordance with metadata encoded as a database schema and partly in the form of free text we define to be *partially structured*[1]. We believe that identifying this type of data as a distinct category will focus the research we argue is needed, clarify its objectives, and lead to new database software products appropriate to what we have identified as a growing and particularly important class of application areas.

The management and exploitation of partially structured data requires database management systems in which modification of both the metdata and instance data is regarded as of equal importance, with both provided for as part of normal database update. The extent of the information which is covered by the schema should then be increased as the understanding of the information and the requirements of the application evolve. Thus, this in the future we will see the schema not as an information specification decided *a priori* at system design time but rather as metadata which evolves and becomes more extensive as information is extracted and analysed from the free text parts.

In its initial form some part, and possibly the greater part, of the information will be in free-text form. As a result of the methodology we envisage, this information will be progressively transformed into new, structured, type and instance information. For example, a statement made to the Police may include text such as "...as I came into the road I saw a man of medium height going into the driveway...". This may lead to the creation of two new subtypes of the person type, witness and suspect, and to the creation of an instance of type person (the "man of medium height" who was seen) which will be assigned as a possible member of both subtypes. The role of the software that we envisage in such a scenario would not be to replace the human analysis and decision making process but to provide tools which materially aid and enhance it.

To succeed, this approach to the management of partially structured information depends firstly upon developing techniques for processing sections of free text in order to identify objects, their classifications, and the relationships among them. We will use techniques derived from research in computational linguistics and natural language understanding for this task. This process will involve some measure of human involvement both in suggesting to, and in confirming or rejecting suggestions from, the software. The semantic and grammatical information thus extracted will be represented in some form of directed graph and/or semantic net although it is not to be expected that *all* of the

[1] This should not be confused with the term *semi-structured data* which now has a currency as meaning data which is "self-describing" and can be represented as a graph [2]

information in the text can be represented in this way and thus the text, or at least some part of it, will continue to be held as such in the database as a necessary part of the information content.

These derived graphs we then aim to semi-automatically transform into new structured type and instance information for integration with the existing structured information in the database. This transformation will require the investigation of methods for identifying semantic similarity between data instances from which new types can be derived to enhance the schema. There may be uncertainty in the assignment of types to instances, in the values of instance attributes, and in the relationships between instances; and there needs to be provision for such uncertainty.

3 Progressing the Research

The aim of this research is to create methods by which the free-form text parts of partially structured information can be progressively clarified and codified so that their information content can be compared with, and become more integrated into, the structured part. In consequence, the structured part becomes of progressively greater value to the application, with the schema itself evolving and becoming itself more informative[2]. Experimental quality software using these methods will be created suitable for experimentation in relevant real world contexts to evaluate the effectivness of the new approach. The research will be progressed in three phases each with a clear objective:

Phase 1: The first objective will be to explore and develop graph-based representations for the semantic and grammatical information extracted from free-form text and for the susequently derived type and instance information. Semi-automated methods will be developed to aid the human in the extraction and analysis of the semantic and grammatical information, and in its transformation to type and instance information.

Phase 2: The second objective will be to design and develop an enhanced Functional Database Programming Language (FDBPL) whose type system will be sufficiently powerful to be able to encode the text-derived data structures and the extraction/transformation methods, as well as the reconciliation and integration of the new structured information with the existing structured information, and the derivation of new schema information from instance information.

Phase 3: The third objective will be to produce a design for a Workbench which will facilitate the initial creation and subsequent progressive evolution of the partially structured databases that we have in mind. The workbench will be used interactively so as to enhance the effectiveness of the expert end-user in using the information in the database to forward the aims of the application[3]. It will have facilities to specify an initial graph-based schema for that part of the information which can be structured *a priori*. The workbench will allow processing of text fragments in order to suggest to the user possible extensions and evolutions of the current database instances, and to test hypotheses for such enhancements from the user. Another important workbench facility will be to aid the recognition of semantic similarity between data instances which could not have been predicted in advance and which can be used to suggest possible schema evolutions to the

[2]Note that this functionality differs from the "schema evolution" facilities traditionally provided by DBMSs and envisaged by previous research into schema evolution, which focus on tools and techniques for migrating database instances and applications to conform to revised database schemas.

[3]In this context we define an end-user to be an expert in the domain of the application able to become readily familiar with graph-based representations of information in their domain.

user. We expect that the FDBPL resulting from Phase 2 will play an important role in realising experimental implementations of the workbench.

4 Timeliness, Relevance and Opportunity

Past research and development in database technology has resulted in data modelling techniques and DBMS software that address successfully many of the major requirements of business and industry. However, the established DBMS technology corresponds to the requirements of the type/instance paradigm whereby common formats for data instances are specified *a priori* in a schema as a result of a process of systems analysis and design, with subsequent database updates not changing significantly the basic model of information.

When this established database technology is used in the kind of application areas that we discussed in Section 1 above the need for an approach which adopts a more flexible and evolutionary approach to database schemas and their specification becomes all too clear. The potential for the use of computer systems in these areas is huge and with the first-generation database applications now being largely solved problems we believe that the time is now ripe to move on to the next generation and to extend the scope and range of database technology in the novel way we propose. Our approach seeks to take advantage of results from work in closely related areas, possibly giving them a new importance, yet retains and develops the solid foundations of current DBMS technology and the benefits it brings.

In Section 1 we briefly described the type of application areas which in our view require the approach we advocate, citing in particular the areas of crime and road accident investigation of which we have direct experience from previous work. There are however many other areas where the computer held information is likely to be partially structured: hospital records, social work case records, educational progress records, and client information in many other professional areas. We intend, however, to confine our experimental work in this project to the two areas of which we already have direct experience and established collaborations since we not only believe them to be of considerable importance in themselves, but that results in these areas will readily generalise to other areas.

To illustrate the type of use of our proposed techniques, consider the Police Force's approach to investigating a serious crime. An important role in any investigation team is that of the "statement reader" whose role is to read a statement as a self-contained piece of information and to identify references to persons, objects and events. This information can then be related to other known or conjectured information, and to references to possibly the same items of information in other statements or in the results of house-to-house inquiries. It can also trigger further or other lines of inquiry. For a particular crime there can be many hundreds of statements. This work is acknowledged to be labour intensive and makes considerable demands on Police resources which are often difficult to meet [1].

With the type of workbench that we envisage, the statement reader would be presented with automatically extracted semantic and grammatical information from a statement, as well as the statement itself. The reader would approve, modify, or supplement this extracted information which would then be transformed and integrated with the existing structured data relating to the particular crime under investigation. This structured data would evolve incrementally as further statements are processed and integrated and reconciled with the existing information about the crime. The totality of information so processed would form a very powerful database to support the investigation in ways well

beyond the capacity of present systems.

Whilst acknowledging that this scenario is ambitious, our experience leads us to believe that it is none-the-less realisable. Although challenging, we believe that the potential benefits of this research are highly significant in extending the range of computer-managed information and its beneficial exploitation. We thus believe that this work deserves further research funding which, if it then proceeds as we envisage, should lead on to ventuer capital funding for practical exploitation.

5 Previous and Related Work

In progressing the research we will need to investigate and review work in areas not normally considered part of database technology which presently address the needs of the relevant application domains, and other work which manipulates and reasons about information. We will also draw upon our our previous research and expertise.

Graph-based Data Models. There will be a need for two kinds of graph-based information representation in this project: (i) the information extracted from free text, and (ii) a graph-based data model for structured type and instance information. Both authors have worked for some years in this area, funded by several SERC/EPSRC projects, specifically within Prof King's TriStarp research project and Dr Poulovassilis' research with Dr Levene on the Hypernode Model. As such we have expertise in graph-based data repositories [21, 50, 22], graph-based data models [23, 37, 4], and graph-based database languages and programming languages [30, 31, 17, 35, 44].

Functional Database Languages. Our plan is to develop an enhanced *functional database programming language* which is sufficiently powerful to encode all of the necessary data structures and algorithms for the proposed research. The authors are international experts in functional database programming languages (FDBPLs). Our contributions in this area include the FDL language, which was the first to integrate the functional data model with functional programming [33, 36, 34], the PFL language which integrated a relational data model with functional programming [45, 39, 42], new query optimisation techniques for FDBPLs [40, 41, 43], active rules for FDBPLs and active rule analysis techniques [6, 5], and the FDL2 language currently under development by the TriStarp Group which will incorporate update functions [28] and second-order querying facilities [4]. Our research will also draw on previous work on using functional formalisms for information integration and as such will have good synergy with recent work in this area by groups at Aberdeen, Manchester, University of Pennsylvania and Uppsala University [18, 20, 7, 32, 10, 19].

Crime Investigation and Road Accident Analysis. The authors have worked on long-term collaborative research projects in these application areas, and these will be the two test beds that we will use for the present research. Prof King led an 18-month project funded by the Home Office in which FDL was used to develop techniques for deriving crime clusters in crime databases relating to burglaries. Dr Poulovassilis has held two collaborative SERC/EPSRC grants with Prof Ben Heydecker at the UCL Centre for Transport Studies (UCLTS) in which PFL was used to develop algorithms for the management and statistical analysis of large volumes of road accident information [52, 51, 16].

Natual Language Understanding. Part of the road accident information analysed at UCLTS was in the form of free text, from which structured information matching the database schema was extracted using natural language understanding (NLU) techniques [53]. That work focused on extracting location information from the text and matched

this directly against a fixed predefined database schema. It did not extend as far as extracting information about events from the road accident reports and to transforming this into an evolving database schema, as we are proposing here.

Handling uncertain information. There may be uncertainty arising from the transformation of information extracted from free-form text into structured information, and accommodating such uncertainly will need to be an inherent part of the graph-based data model that we develop. Our relevant expertise here is in extending functional data models to handle incomplete information [48, 49]. Dr Daniel Stamate, currently a Marie-Curie Research Fellow in the department, also has expertise in this area [14, 24] and is likely to contribute in this aspect of the project.

Schema equivalence and schema integration. There may be semantic conflicts between the existing types/instances and the structured information derived from free-form text which will need to be handled and resolved. Our relevant expertise here is Dr Poulovassilis' recent work with Dr McBrien on equivalence, transformation and integration of database schemas, based an underling graph-based representation of higher-level schema constructs [38, 27, 26]. Drs McBrien and Poulovassilis have recently been awarded an EPSRC grant under the second DIM call (no. GR/N 35915) to continue this work in the areas of mediator/wrapper generation, schema improvement, and global query optimisation for heterogeneous databases, and that project will have a good synergy with the present research.

The proposed work will draw from several other major research areas, of which **information retrieval**, **conceptual models**, **semi-structured data models**, **text mining** and **computational linguistics** are the most relevant. A good overview of the state-of-the-art in these areas can be found in the proceedings of the SIGIR, ICCS, WebDB, WWW, and ACL conference series, respectively.

The traditional aim of information retrieval is to select documents that satisfy users' criteria using concept-matching techniques. Our aim has a different focus in that we will be extracting grammatical structure as well as concept references from whole collections of documents. The work on conceptual models will be relevant to the way that we represent this grammatical information. Work has recently been done on integrating IR and database languages, for example in probabilistic query languages and algebras [13] and this work will be relevant to our handling of uncertainty in the structured information derived from the free text.

The graph-based data model that we develop to represent structured information may well have a close relationship to semi-structured data models and the emerging XML standards, the relationship of which to structured data models is work we currently have in progress (see `http://www.dcs.bbk.ac.uk/~ap/pubs /tr050800.ps` and `http://www.dcs.bbk.ac.uk/TriStarp/resrep0820001.html`). Particularly relevant here is the work on extracting schema information from semi-structured data (surveyed in [2]) of which the approach of Nestorov *et al.* [29] is the most relevant to our proposed research in that it allows approximate, i.e. possibly imperfect, typing of objects which may have multiple roles. However, we are again seeking to achieve something beyond this in not undertaking simple graph-matching of database instances in order to derive schema information, but more sophisticated analysis of semantic similarity between database instances. Relevant work in text mining includes the Google search engine which extracts information of fixed, known format from the WWW [8, 9], the WHIRL system which uses text-matching methods from information retrieval to retrieve and integrate information from the WWW [12], *focussed crawling* which seeks web pages that are relevant to a specified set of topics [11], and NoDoSE [3] which compares text against a user-specified

structure. However, our proposed approach goes beyond this work in that text will be analysed using a much more sophisticated natural language lexicon. Finally, work in NLU has generally focused on text analysis, representation and manipulation (see for example discussions in [15, 25]) rather than on utilising information extracted from text for the kinds of database applications that we have in mind.

6 Conclusions

This research report has identified and discussed an important class of applications which need better and more appropriate database management software products than are currently available. We have indicated how such new products should be engineered and identified the research necessary for their development and realisation. We would encourage those interested in this work and its outcome, and with potential applications, to follow progress on our website, assist aand participate. For intending research students we believe there to be a number of themes which could provide subjects for interesting PhD theses and which would make a significant contribution to the work. There are also aspects which could lead to very good MPhil dissertations.

References

[1] The Stephen Lawrence Inquiry; report by Sir William MacPherson of Cluny. *CM 4262-I*, February 1999.

[2] S. Abiteboul, P. Buneman, and D. Suciu. *Data on the Web - From Relations to Semistructured Data and XML*. Morgan Kaufmann, 2000.

[3] B. Adelberg. NoDoSE - A tool for semi-automatically extracting structured and semistructured data from text documents. In *Proc. SIGMOD'98*, pages 283–294, 1998.

[4] R. Ayres and P.J.H. King. Querying graph databases using a functional language extended with second order facilities. In *Proc. BNCOD-14*, volume 1094 of *LNCS*, pages 189–203. Springer-Verlag, 1996.

[5] J. Bailey and A. Poulovassilis. An abstract interpretation framework for termination analysis of active rules. In *To appear in Proc DBPL'99*. Springer-Verlag.

[6] J. Bailey and A. Poulovassilis. Abstract interpretation for termination analysis in functional active databases. *Journal of Intelligent Information Systems*, 12(2/3):243–273, 1999.

[7] P.G. Baker, A. Brass, S. Bechhofer, C.A. Goble, N.W. Paton, and R. Stevens. TAMBIS: Transparent access to multiple bioinformatics information sources. An Overview. In *Proc. 6th International Conference on Intelligent Systems for Molecular Biology, ISMB98*, pages 25–34, 1998.

[8] S. Brin. Extracting patterns and relations from the world wide web. In *Proc. WebDB*, pages 172–183, 1998.

[9] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proc. WWW7*, pages 107–117, 1998.

[10] P. Buneman, J. Crabtree, S. Davidson, V. Tannen, and L. Wong. Biokleisli. In S.Letovsky, editor, *Bioinformatics*. Kluwer, 1998.

[11] S. Chakrabarti, M. van den Berg, and B. Dom. Distributed hypertext resource discovery through examples. In *Proc. VLDB'99*, pages 375–386, 1999.

[12] W.W. Cohen. Integration of heterogeneous databases without common domains using queries based on textual similarity. In *Proc. SIGMOD'98*, pages 201–212, 1998.

[13] E. Moss (ed.). Special issue on integrating text retrieval and databases. *Data Engineering Bulletin*, 19(1), 1996.

[14] G. Grahne, N. Spyratos, and D. Stamate. Semantics and containment with internal and external conjunctions. In *Proc. ICDT'97*, volume 1186 of *LNCS*, pages 71–82. Springer-Verlag, 1997.

[15] M. Hearst. Untangling text data mining. In *Proc. ACL'99*, 1999.

[16] B. Heydecker, C. Small, and A. Poulovassilis. Deductive databases for transport engineering. *Transportation Research, Part C - Emerging Technologies*, 3(5):277–292, 1995.

[17] S. Hild and A. Poulovassilis. Implementing Hyperlog, a graph-based database language. *Journal of Visual Languages and Computing*, 7:267–289, 1996.

[18] K. Hui and P.M.D. Gray. Constraint and data fusion in a distributed information system. In *Proc. BNCOD-16*, volume 1405 of *LNCS*, pages 181–182. Springer-Verlag, 1998.

[19] V. Josifovski and T. Risch. Functional query optimization over object-oriented views for data integration. *Journal of Intelligent Information Systems*, 12(2/3):165–190, 1999.

[20] G.J.L. Kemp, C.J. Robertson, P.M.D. Gray, and N. Angelopoulos. CORBA and XML: Design choices for database federations. In *Proc. BNCOD-17*, volume 1832 of *LNCS*, pages 191–208. Springer-Verlag, 2000.

[21] P.J.H. King, M. Derakhshan, A. Poulovassilis, and C. Small. Tristarp - an investigation into the implementation and exploitation of binary relational storage structures. In *Proc. BNCOD-8*, pages 64–84. Pitman, 1990.

[22] J.K. Lawder and P.J.H. King. Using space-filling curves for multi-dimensional indexing. In *Proc. BNCOD-17*, volume 1832 of *LNCS*, pages 20–35. Springer-Verlag, 2000.

[23] M. Levene and A. Poulovassilis. An object-oriented data model formalised through hypergraphs. *Data and Knowledge Engineering*, 6:205–224, 1991.

[24] Y. Loyer, N. Spyratos, and D. Stamate. Computing and comparing semantics of programs in four-valued logics. In *Proc. MFCS'99*, volume 1672 of *LNCS*, pages 59–69. Springer-Verlag, 1999.

[25] P. Martin and P. Eklund. Embedding knowledge in web documents: Cgs versus xml-based metadata languages. In *Proc. Proc ICCS'99*, pages 230–246, 1999.

[26] P.J. McBrien and A. Poulovassilis. Automatic migration and wrapping of database applications — a schema transformation approach. In *Proc. ER'99*, volume 1728 of *LNCS*, pages 96–113. Springer-Verlag, 1999.

[27] P.J. McBrien and A. Poulovassilis. A uniform approach to inter-model transformations. In *Proc. CAiSE'99*, volume 1626 of *LNCS*, pages 333–348. Springer-Verlag, 1999.

[28] P.F. Meredith and P.J.H. King. Scoped referential transparency in a functional database language with updates. In *Proc. BNCOD-16*, volume 1405 of *LNCS*, pages 134–148. Springer-Verlag, 1998.

[29] S. Nestorov, S. Abiteboul, and R. Motwani. Extracting schema from semistructured data. In *Proc. SIGMOD'98*, pages 295–306, 1998.

[30] A. Papantonakis and P.J.H. King. Gql, a declarative graphical query language based on the functional data model. In *Proc. Advanced Visual Interfaces (AVI'94)*, pages 113–122, 1994.

[31] A. Papantonakis and P.J.H. King. Syntax and semantics of Gql, a graphical query language. *Journal of Visual Languages and Computing*, 6:3–25, 1995.

[32] N.W. Paton, R. Stevens, P. Baker, C.A. Goble, S. Bechhofer, and A. Brass. Query processing in the TAMBIS bioinformatics source integration system. In *Proc. SSDBM*, pages 138–147, 1999.

[33] A. Poulovassilis. FDL : an integration of the functional data model and the functional computational model. In *Proc. BNCOD-6*, pages 215–236. C.U.P., 1988.

[34] A. Poulovassilis. The implementation of FDL, a functional database language. *The Computer Journal*, 35(2):119–128, 1992.

[35] A. Poulovassilis and S. Hild. Combining declarative querying and browsing in a visual database language. *To appear in IEEE Knowledge and Data Engineering.*

[36] A. Poulovassilis and P.J.H. King. Extending the functional data model to computational completeness. In *Proc. EDBT-90*, volume 416 of *LNCS*, pages 75–91. Springer-Verlag, 1990.

[37] A. Poulovassilis and M. Levene. A nested-graph model for the representation and manipulation of complex objects. *ACM Trans. on Information Systems*, 12(1):35–68, 1994.

[38] A. Poulovassilis and P.J. McBrien. A general formal framework for schema transformation. *Data and Knowledge Engineering*, 28(1):47–71, 1998.

[39] A. Poulovassilis and C. Small. A functional programming approach to deductive databases. In *Proc. 17th VLDB*, pages 491–500, 1991.

[40] A. Poulovassilis and C. Small. Investigation of algebraic query optimisation for database programming languages. In *Proc. 20th VLDB*, pages 415–426, 1994.

[41] A. Poulovassilis and C. Small. Algebraic query optimisation for database programming languages. *The VLDB Journal*, 5(2):119–132, 1996.

[42] A. Poulovassilis and C. Small. A domain-theoretic approach to integrating functional and logic database languages. In *Proc. 19th VLDB*, pages 416–428, 1996.

[43] A. Poulovassilis and C. Small. Formal foundations for optimising aggregation functions in database programming languages. In *Proc DBPL'97*, volume 1369 of *LNCS*, pages 299–318. Springer-Verlag, 1997.

[44] P.J. Rodgers and P.J.H. King. A graph-rewriting visual language for database programming. *Journal of Visual Languages and Computing*, 8:641–674, 1997.

[45] C. Small and A. Poulovassilis. An overview of PFL. In *Proc. DBPL'91*, pages 96–110. Morgan Kaufmann, 1991.

[46] J.F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, Reading, MA, 1984.

[47] J.F. Sowa. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

[48] D.R. Sutton and P.J.H. King. The integration of modal logic and the functional data model. In *Proc. BNCOD-10*, volume 618 of *LNCS*, pages 156–174. Springer-Verlag, 1992.

[49] D.R. Sutton and P.J.H. King. Incomplete information and the functional database model. *The Computer Journal*, 38:31–42, 1995.

[50] E. Tuv, A. Poulovassilis, and M. Levene. A storage manager for the hypernode model. In *Proc. BNCOD-10*, volume 618 of *LNCS*, pages 59–77. Springer-Verlag, 1992.

[51] J. Wu and L. Harbird. A functional database for road accident analysis. *Advances in Engineering Software*, 26(1):29–43, 1996.

[52] J. Wu and B. Heydecker. A knowledge-based system for road accident remedial work. *Computing Systems in Engineering*, 4(2-3):337–348, 1993.

[53] J. Wu and B. Heydecker. Natural language understanding in road accident data analysis. *Advances in Engineering Software*, 29(7-9):599–610, 1998.