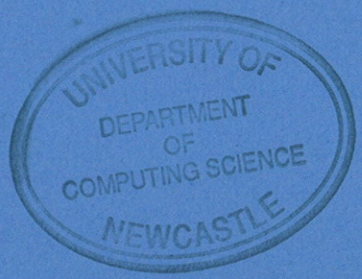
UNIVERSITY OF
NEWCASTLE UPON TYNE

University of Newcastle upon Tyne

COMPUTING SCIENCE



Data Management of On-line Information Systems

B.N. Rossiter and M.A. Heather

TECHNICAL REPORT SERIES

No. 406 December, 1992

TECHNICAL REPORT SERIES

No. 406 December, 1992

Data Management of On-line Information Systems

B.N. Rossiter and M.A. Heather

Abstract

Large commercial information systems still mainly concentrate on secondary and tertiary information rather than the goal-information actually sought by end-users. In this report, the nature of goal-information is explored and means by which it may be handled by database systems are discussed. A comparison is made of promising developments such as object-oriented databases and dynamic semantic data models with more traditional approaches using relational databases and free text retrieval systems.

Printed and published by the University of Newcastle upon Tyne,
Computing Science, Claremont Tower, Claremont Road,
Newcastle upon Tyne, NE1 7RU, England.

Bibliographical details

ROSSITER, Brian Nicholas

Data Management of On-line Information systems
[By] B.N. Rossiter and M.A. Heather

Newcastle upon Tyne: University of Newcastle upon Tyne: Computing Science, 1992.

(University of Newcastle upon Tyne, Computing Science, Technical Report Series, no.406)

Added entries

UNIVERSITY OF NEWCASTLE UPON TYNE.
Computing Science. Technical Report Series. 406
HEATHER, Michael A.

Abstract

Large commercial information systems still mainly concentrate on secondary and tertiary information rather than the goal-information actually sought by end-users. In this report, the nature of goal-information is explored and means by which it may be handled by database systems are discussed. A comparison is made of promising developments such as object-oriented databases and dynamic semantic data models with more traditional approaches using relational databases and free text retrieval systems.

About the author

Nick Rossiter is a lecturer in the Department of Computing Science with particular interests in databases and system analysis.

Michael Heather is senior lecturer in law where he has been responsible for computers and law since 1979.

Suggested keywords

DATABASE MANAGEMENT SYSTEMS
END-USER
FULL TEXT
GOAL INFORMATION
OBJECT-ORIENTED PARADIGM
RELATIONAL MODEL
SEMANTIC MODELS

Suggested classmarks (primary classmark underlined)

Dewey (18th): 001.6442 029.7
U.D.C. 681.322.06 651.838.8

Data Management of On–line Information Systems

B.N. Rossiter
Computing Science
University of Newcastle

M.A.Heather
Sutherland Building
University of Northumbria

November 1992

Abstract

Large commercial information systems still mainly concentrate on secondary and tertiary information rather than the goal–information actually sought by end–users. In this report, the nature of goal–information is explored and means by which it may be handled by database systems are discussed. A comparison is made of promising developments such as object–oriented databases and dynamic semantic data models with more traditional approaches using relational databases and free text retrieval systems.

About the author

Nick Rossiter is lecturer in the Department of Computing Science with particular interests in databases and systems analysis.

Michael Heather is senior lecturer in law where he has been responsible for computers and law since 1979.

Suggested Keywords

Database management systems, full text, goal information, end–user, relational model, object–oriented paradigm, semantic models.

Suggested classmarks

Chapter 1

Introduction

The shift from printed paper to electronic systems has still not fully penetrated every sector of information. About 90% of CD–ROM and on–line systems give short reference rather than full information. There still remains the hardly–touched large information systems where end–users come away with the fine detail of goal–information not with mere abstracted knowledge or pointers to fuller sources elsewhere. The very success of the abstracting services is leaving this class of user frustrated rather than satisfied, in knowing now that a wealth of sources exist but which may not be easily accessible.

Elaborate data management systems are needed to provide the high functionality required for structuring, manipulating and maintaining the data with the necessary integrity. Results are presented of an investigation into current database models to show the facilities required to support the more sophisticated interfaces now demanded of information providers. End–users require access transparently to goal–information in a highly organised state.

Chapter 2

The Move to Goal–information

Information defies absolute classification but from the point of view of users of information systems, there are two basic types of importance:

- Goal–information satisfies the ultimate informational needs of the end–user;
- Indicative information has limited informational content. The main aim is to provide an awareness of the existence of some information. It cannot provide full satisfaction and often merely points in the direction of goal-information.

Thus items sometimes classified as primary information like newspapers or patents are not really goal–information. A newspaper story can usually only be taken as a superficial account of events and if the information in the report is to be relied on for any serious purpose, further research is needed to verify the sources. Likewise patents are often drafted more to conceal information than to enlighten. The goal–information is contained in the patent owners original plans, drawings, specifications, test results, prototype constructions, description of chemical processes and techniques, etc. There will probably be a requirement for different levels of restriction for different levels of access.

The shift from printed paper to electronic systems has only just penetrated the indicative sector of information. About 90% of CD–ROM and on–line systems give short reference rather than full information. Familiar examples of indicative information are abstracts, indexes, patents, financial summaries, newspapers, statistics and bibliographies. The sector of goal–information remains hardly touched and is only represented commercially by small examples of electronic books like the multi–version bible, atlases, dictionaries, small encyclopaediae, language courses, etc. Full scientific papers, specifications, raw data in medicine, environmental sciences like seismology, ecology, etc and the social sciences are not yet widely available.

This contrast is also reflected in the division of those who use information. There are the end–users who are the ultimate consumer of the goal–information

and there are the intermediaries or mid–users, often those whose profession is to assist in the provision of information, who are more concerned with the indicative class of information. The end–user's preference is to reach the goal–information transparently without the need to consult bibliographies, etc.

Partly for technological reasons, partly from the not unconnected reason of market forces, interest has concentrated on the indicative information which is easier to handle as it is smaller in volume and more readily structured but because of its very nature is mainly consulted by mid–users. Indeed there has been such a growth of what is often known as secondary information that it has generated a strong tertiary information industry consisting of directories devoted to the service industries.

This all leads to management information problems but especially in handling goal-information. Commercial provision, where end–users come away with the fine detail of goal–information not with mere abstracted knowledge or pointers to fuller sources elsewhere. still remains rare for the large information systems. The very success of the abstracting services is leaving this class of user frustrated rather than satisfied, in knowing now that a wealth of sources exist but which may not be easily accessible.

However this does not mean that there will be an immediate up–take by end–users for goal–information [Blair & Maron 1985]. Mid–users and information providers are well aware of the many advantages of electronic information systems and may often wonder why end–users do not make much greater use of these services. The answer may be that end–users have sophisticated needs but do not have the same sophisticated interest as mid–users have in the process of information retrieval for its own sake. Before there can be an explosion in goal-information services there may be need for further heed to be given to the technological facilities for end-users.

On the one side, there is the heterogeneous complex nature of goal information which requires very much more elaborate processing and storage in itself. Then also because it has to be of direct use by this other class of the end–user, any system has to be doubly–better to provide transparent functionality at the appropriate level. In particular, the functionality has to operate at the goal–information level, it is not sufficient to rely on the capabilities of an interface, however attractively it may be presented.

An example of the power of the technology needed by an end–user is the problem of estimating the relevance of information presented in surrogate form. This is often difficult. Precision is greatly enhanced if the end–information can be inspected at the time for relevance. An end–user may like a system to be able to provide direct automatic input into the full text goal–information at exactly the point of interest identified by an awareness document like a bibliographic abstract.

1. Design of STRUCTURE for holding text
   - unlimited size of fields and records
   - symbolic identification of records
   - data models (hierarchical and non--hierarchical)
   - ability to retain un-normalized data
   - dynamic control of unit size
   - generalization and specialization
   - acceptance of standards (e.g. SGML, ODA, MARC)
2. RETRIEVAL
   - fast
   - non-procedural interactive languages
   - words + phrases in text (context and proximity)
   - assistance with query construction
   - iterative searching, result stack management
   - keywords (free and controlled vocabulary, thesauri)
   - 'formatted' data
   - identifiers of text (symbolic key)
3. TEMPORAL management with consistent updating
   - in-place modification, addition of data, archiving
   - dynamic behaviour - control of document life cycle
   - version management
   - concurrent access
   - value inheritance for natural data loading
4. MULTI-MEDIA . integration of text and other data (unified model)
5. INTEGRITY . protection against hardware failures
   - referential and value
6. VIEWS .
   - derived structures
   - parallel texts
7. Various formats for DISPLAY and OUTPUT
   - human (reading and listening, etc)
   - machine-machine (wp, mark-up)
   - modes for users with special needs
8. NAVIGATION through texts following conceptual paths
   - referential transparency (hypertext)
   - reference to external sources including secondary and tertiary material
   - trail maintenance
9. SECURITY . whole file, designated fields, data driven
10. SEMANTICS . parsing, predicate logic, machine translation
    - cognitive textual types
11. Textual ANALYSIS
    - function integrated with data
    - word co--occurrences, frequency lists, distribution
    - statistical tests e.g. sentence length

Figure 2.1: Required Functionality for DBMS from Users' Viewpoint

Chapter 3

Goal–information DBMS

A data store under the control of a Database Management System (DBMS) is searchable and updateable under the control of a data model which defines data structures, retrieval facilities and rules for maintaining consistency and integrity. It is therefore suggested that DBMS are a good starting point for satisfying user requirements. The issue is whether standard formal database models like the relational or specialised software such as free text retrieval or hypertext systems are sufficient to meet the required standards and whether future products such as semantic and object-oriented databases offer much better prospects.

We now consider in more detail the ability to handle the complex requirements shown in Figure 2.1 by existing technology and the kind of techniques that are likely to be available in the near future. First we emphasise that none can completely satisfy the requirements of Figure 2.1. The capabilities of present systems for handling text as complex objects are summarized in Figure 3.1.

In free text retrieval systems, there is an emphasis on handling large records with flexible fast retrieval methods. But there is little flexibility in viewing the data in different ways - the unit size is fixed, aggregation of records to form larger units is difficult and few abstractions such as the generalisation and specialisation of inheritance are available to ease the task of handling large numbers of data types where there are many common properties but each type may have its own peculiarities. Further, it is not easy to build a hypertext system on top of a free text retrieval system : the necessary symbolic identifiers and cross-referencing facilities do not exist.

Specialised software has been developed to handle the hypertext requirement. Such software provides attractive interfaces to the user but many systems lack flexibility. It is not sufficient to provide a 'hard-wired' set of buttons and links. There is the need to provide a continuing system for interpolation, updates, correction, etc. Because of the need for a flexible cross–referencing system, such hypertext requirements are in the natural domain of database systems.

Relational databases cope well in some of the areas in which the free text retrieval and hypertext systems are weak: views enable many different variants

	Free Text (1)	Relational stand. (2)	Relational extend. (3)	Semantic static (4)	Semantic dynamic (5)	Object-oriented (6)
1. Design of STRUCTURE for holding text						
. unlimited size of fields and records	yes	poss	poss	yes	-	yes
. symbolic identification of records	limited	yes	yes	yes	yes	yes
. data models						
. hierarchical		yes	yes	yes	yes	yes
. non-hierarchical (shared sub-obj.)		no	diff	yes	yes	yes
. ability to retain un-normalized data		yes	no	no	no	yes
. dynamic control of unit size (aggreg)		no	diff	diff	yes	diff
. generalization and specialization (inher.)		no	no	no	yes	yes
. acceptance of standards (SGML, ODA, MARC)	yes	no	no	no	no	no
2. RETRIEVAL						
. fast	yes	no	yes	no	-	yes
. non-procedural interactive languages	yes	yes	yes	no	no	diff
. closure	no	yes	yes	no	no	no
. words + phrases in text						
. context	yes	yes	yes	no	yes	yes
. proximity matching	yes	no	yes	no	yes	yes
. assistance with query construction	yes	no	no	no	no	no
. iterative searching, result stack facil.	yes	no	no	no	no	no
. keywords						
. free vocabulary	yes	yes	yes	no	yes	yes
. controlled vocabulary (thesauri, stop)	yes	yes	yes	no	yes	yes
. 'formatted' data	limited	yes	yes	no	yes	yes
. identifiers of text (symbolic key)	limited	yes	yes	no	yes	yes
3. TEMPORAL management with consistent updating						
. in-place modification, addition, archiving	yes	yes	yes	no	yes	yes
. dynamic behaviour - control of doc. life	no	no	no	no	yes	yes
. version management	no	no	no	no	yes	yes
. concurrent access	yes	yes	yes	no	-	diff
. value inheritance for natural data loading	no	no	no	yes	yes	yes
4. MULTI-MEDIA						
. integrate text + other data - unified model	no	no	poss	poss	poss	poss
5. INTEGRITY						
. protection against hardware failures	yes	yes	yes	no	-	yes
. referential	no	yes	yes	no	yes	yes
. value	yes	yes	yes	no	yes	yes
6. VIEWS						
. derived structures	no	yes	yes	no	yes	yes
. parallel texts	no	no	yes	no	poss	poss
7. Various formats for DISPLAY and OUTPUT						
. human (reading and listening, etc)	yes	yes	yes	no	-	yes
. machine-machine (wp, mark-up)	yes	yes	yes	no	-	yes
. modes for users with special needs	yes	poss	poss	no	-	poss
8. NAVIGATION through texts following conceptual paths						
. referential transparency (hypertext)	no	yes	yes	no	yes	yes
. trail maintenance	no	yes	yes	no	yes	yes
9. SECURITY						
. whole file, designated fields, data driven	yes	yes	yes	no	yes	yes
10. SEMANTIC factors						
. parsing	no	no	poss	no	poss	poss
. predicate logic, machine translation	no	no	poss	no	poss	poss
. cognitive textual types	no	no	poss	no	poss	poss
11.Textual ANALYSIS						
. function integrated with data	no	no	no	no	yes	yes
. word frequency, distrib. and co-occurrences	yes	no	yes	no	poss	poss
. statistical tests e.g. sentence length	yes	no	yes	no	poss	poss

Figure 3.1: Capabilities of Present Systems for Handling Text

Notes to Figure 3.1

1. Free text retrieval systems, e.g. Basis, BRS/Search and Status, with relatively limited data structuring capabilities but powerful retrieval facilities for searching on combinations of terms in variable context.

2. Relational systems, e.g. Oracle, Ingres and DB2, with data held in a flattened (normalised) form in tables and manipulated with a set-theoretic query language. Some semi–relational systems such as dBase exhibit similar characteristics. Unit sizes (rows in tables) are determined by grammatical or semantic fragments typically phrases, clauses, sentences, paragraphs, chapters.

3. Extended relational systems, as emulated with Spires in the project investigating parallel versions of the bible [Heather & Rossiter 1990] where words are the basic units of data and other units are constructed dynamically at run–time.

4. Static semantic models, e.g. the E-R extended model [Chen 1976], with the basic concepts of entities, relationships and attributes, and abstractions of inheritance and aggregation.

5. Dynamic semantic models, e.g. Taxis [Mylopoulos 1980], similar to static models but with the addition of features to represent behaviour and life–cycles.

6. Object–oriented database systems, e.g. O_2, Gemstone and Versant, which can be viewed as persistent object stores of object-oriented programming languages such as C++. Through evolutionary convergence, there are many similarities between this category and the previous one.

Significance of entries:

- **yes** : achieved with relative ease as a natural feature of the approach;
- **limited** : partial achievement of facility;
- **poss[ible]** : not achieved to date but there is no reason to doubt that with effort the facility could be effected;
- **diff[icult]** : not a natural feature of the approach but the facility can be achieved with considerable effort;
- **no** : outside the scope of the approach;
- **–** : no information.

of the basic data structures to be presented to the end–user and hypertext is feasible as symbolic identifiers can be defined and manipulated using a standard relational query language such as SQL. However, perhaps frustratingly, relational databases are not simply an incremental advance on free text systems. They often, in a retrograde way, suffer from arbitrarily low limits on record sizes. Searching on patterns and for character strings within a text usually involves a serial search giving very poor response times.

A more fundamental criticism of relational databases is that they are too naive in their ability to capture the complexities of textual information. The restriction of users to one type of data structure, a table, is too limiting and in large applications with many different text–types the user finds it difficult to gain any perspective on the links between tables and the overall structure. Mastering such complexity is the province of the semantic data models and object–oriented databases.

Semantic data models and object-oriented data bases may have very different origins. The former represent an attempt to extend the modelling capability of databases while retaining the mathematical basis, whereas the latter represent good practice in program design on a proven engineering basis. Nevertheless there are many similarities in their usage and their practical status. Both aim to provide abstractions to handle complexity such as inheritance and aggregation as described earlier. These abstractions enable the user to manipulate data structures which are complex and/or large by providing a natural rationalisation.

Both approaches also concentrate on directed–graph representations of data structures, which can be readily represented in diagrammatic form to facilitate understanding by end–users. The idea of a 'map' is somtimes used today to describe the visualisation for users of a configuration space. In the object–oriented paradigm and in some semantic data models such as Taxis, the behaviour of objects and their structure is covered by a single unified model.

However, neither approach can offer a panacea at present [Tsichritzis & Nierstrasz 1988]. Both suffer from implementation difficulties, owing to the complexity of their data structures, which means that it is only recently that commercial versions have started to become available. Individual weaknesses are the difficulty of achieving the aggregation abstraction in some object–oriented systems. The amalgamation of objects militates against the spirit of the paradigm. It is difficult to provide user–friendly non-procedural query languages because of the imperative nature of the paradigm.

Then there is the difficulty of defining behavioural characteristics in some semantic data models which are after all basically only data structuring tools. For instance the Entity–Relationship model does not possess characteristics beyond a static model and cannot perform as a dynamic text management system.

3.1 Present Position

To conclude, the requirements of the advanced end–user are seen to be rather too complex for any system around at the present time and it cannot be assumed that forthcoming technology such as semantic and object-oriented databases will be able to handle the necessary features. In particular there is a need for the development of semantic data models which integrate static and dynamic characteristics. As hinted above this requires a rigorous theoretical approach at the abstract level.

Chapter 4

References

Blair & Maron 1985 Blair, D C, & Maron, M E, (1985), An Evaluation of Retrieval Effectiveness for a Full-Text Document-Retrieval System, CACM **28** 289-299.

Chen 1976 Chen, P P–S, (1976), The Entity–Relationship Model – towards a unified view of data, ACM TODS **1**(1) 9–36.

Heather & Rossiter 1990 Heather, M A, & Rossiter, B N, (1990), Syntactical Relations in Parallel Text, in: Proc. 15th Int. ALLC Conference, ed. Chouka, Y, Jerusalem 1988, 197-214.

Mylopoulos, Bernstein & Wong 1980 Mylopoulos, J, Bernstein, P A, & Wong, H K T, (1980), A Language Facility for Designing Database–Intensive Facilities, ACM TODS **5** 185-207.

Rossiter & Heather 1990 Rossiter, B N, & Heather, M A, (1990), Strengths and Weaknesses of Database Models for Textual Documents, Proceedings EP90, ed. R. Furuta. Cambridge 125-138.

Rossiter, Sillitoe & Heather 1990 Rossiter, B N, Sillitoe, T J, & Heather, M A, (1990), Database Support for very large Hypertexts, EP-odd **3**(3) 141-154.

Tsichritzis & Nierstrasz 1988 Tsichritzis, D C, & Nierstrasz, O M, (1988), Fitting Round Objects into Square Databases, ECOOP'88 Proceedings, in: Lecture Notes in Computing Science, Springer-Verlag **322** 283-299.