Conceptual Data Modelling for Geographical Information Systems

A thesis presented in partial fulfilment of
the requirements for the degree of

Master of Arts (Information Systems)
at Massey University

Erzsebet Bekesi

1994

Abstract

This thesis sets out to find an answer to the question: does an appropriate conceptual data model exist for the practitioners of Geographical Information Systems database design? It aims to investigate and answer the question by:

- Finding a workable data model to solve a database design problem (Manawatu-Wanganui Regional Council, Palmerston North, Natural Resources Management, Groundwater Section database).

- Analysing the user's data requirements and producing a feasible conceptual schema.

Usage of Geographical Information Systems applications is a recognised need in a growing number of organisations in New Zealand, but many factors block the way of this relatively new technology. One of these factors is the lack of well-designed databases to support the data needs of these non-traditional applications. One school of thought adopts general data modelling techniques for every database design problem, another group of researchers suggests that specialised data models are necessary to model data in various problem domains.

This thesis summarises the "specialities" pertaining to the GIS database domain. The most important are the special data needs of GIS applications and the problem of the placement of spatial data models in the traditional taxonomy of database models. It chooses the objectives of conceptual data modelling as the evaluation criteria which the selected data model must satisfy i.e. to model reality and to form the basis for database schema design.

This thesis reviews a group of published papers, selected from proponents of the entity-relationship and the object-oriented data modelling paradigms and the applications of these data modelling techniques in a spatial context. It compares various extensions to the original entity relationship model, and a comparison of the main data modelling paradigms is included. Data modelling shortcomings encountered in the literature are also summarised. The literature reviewed concludes that not appreciating the conceptual data modelling objectives leads to unsatisfactory conceptual database design.

The selected data model, the spatially extended entity relationship (SEER) model is described and applied to the database design problem of a local authority to produce conceptual schemas. Findings are summarised and issues for future research are identified.

Conclusions reached are: further evaluative work on the applied spatially extended entity relationship (SEER) model would be useful and clear directions are essential for practitioners showing the guiding principles of conceptual data modelling in a spatial context.

Acknowledgements

I wish to acknowledge the financial assistance provided by the Department of Information Systems at Massey University for my masterate studies.

I thank my supervisors for their work: Elisabeth G. Todd, Lecturer, for pointing me the right direction during my studies at Massey University and Roger Tagg, Senior Lecturer, for working through several drafts of this thesis with me.

I thank Barry Jackson, Senior Lecturer for his support at critical times during the year.

A special thanks goes to Dr Peter G. Firns, Lecturer, Department of Information Science, University of Otago for reading and commenting on the case study of this thesis, and for his helpful suggestions in stressful moments of the last few months. Peter, I wish you a lot of success in your future research work.

Thanks Shamus, Shelley, David and Paul for sharing laughs and disappointments in room 1.38. I must thank everybody in the Research Lab for the supportive and pleasant working environment. A special thanks goes to Raymond for translating my thesis into a readable English.

I thank my husband, Gabor Bekesi, underground water scientist, Manawatu-Wanganui Regional Council for his input to the case study, and I dedicate this thesis to our daughter Jane Ingrid.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction and Objectives

1.1 Research Motivation and Problem Statement

In the information age data is a significant and expensive corporation asset. The "database approach" for data management has many well publicised benefits (McFadden and Hoffer, 1988; Elmasri and Navathe, 1989; Date, 1991). The integration of a corporate database allows its contents to be controlled while providing:

- valid information for problem solving and decision making
- for data independence from the programs that use it.

The realisation of these goals is within the reach of businesses and organisations with the help of accurately designed databases that support the required applications. The role of database design, as a prelude activity in the attainment of any database is universally recognised in the business community.

There is a growing need to use database systems to support applications beyond the traditional transaction driven or data retrieval business applications. Currently these non-traditional applications include computer-aided design, software engineering and manufacturing systems, knowledge-based systems, multimedia systems and Geographical Information Systems (GIS). Geographical Information Systems share the trait of traditional data retrieval systems, data is collected and stored in the system before the system can be utilised, but these systems possess many distinctive characteristics, an issue which will be discussed in chapter 2.

Geographical Information Systems have been generating great interest world-wide and there is a growing usage of GIS technology in New Zealand regional and city councils (Fraser and Todd, 1994). Although it is apparent from the literature that methodologies such as the system development life cycle concept[1] are being advocated to GIS users (Chambers, 1989; refer to figure 1.1), there seems to be widespread agreement in the GIS community (Goodchild et al., 1992; Laurini and Thompson, 1992) that among other reasons[2], the real potential of this relatively new technology has not yet been realised due to a lack of well designed databases.

---

[1]Conceptual design is part of this methodology (refer to figure 1.1).
[2]lack of funds, lack of qualified staff etc. (Tomlinson, 1991)

Many researchers (Brodie, 1984; Bedard, 1989) believe that the reason for not using formal methods of database design in the acquisition of Geographical Information Systems is more complex than it was initially thought. Conceptual data modelling methods for the design of databases supporting GIS applications is the subject of ongoing research (Bedard and Paquette, 1989; Armstrong & Densham, 1990; Worboys et al., 1990 etc.), but maybe the real hindrance to their usage is, it is contended by the researchers, the absence of adequately adjusted conceptual data models for the design of GIS databases.

This situation raises the question: does an appropriate tool[3] (i.e. conceptual data model) exist for the practitioners of GIS database design? This thesis aims to investigate and answer this question by:

- Finding a suitable data model to solve a database design problem (Manawatu-Wanganui Regional Council, Palmerston North, Natural Resources Management, Groundwater Section database).

- Analysing the user's data requirements and producing a feasible conceptual schema.

1.2 Background to Conceptual Data Modelling

This section provides a preliminary introduction into the discussion of conceptual data modelling for any type of information system. First a summary of the basic information needed in this domain is given, followed by some basic definitions and finally the various conceptual data modelling paradigms are discussed.

[3]The word "tool" is defined in this thesis as "a means to an end" and **not** as it is sometimes defined in information engineering terminology (i.e. tools are "*implements employed by people to reduce manual effort in accomplishing work*" Case, 1986 cited by Firns, 1990b:430).

1.2.1 General Background for All Types of Database Design

A database is an interrelated collection of data files with the purpose of supporting the data needs of multiple applications. Before a database is populated, its structure must be carefully designed and built, generally through three phases: the conceptual, the logical and the physical database design phases.

The three design phases have three corresponding data models. These models are tools which database designers use to produce instances of the three different abstraction levels' data models. Figure 1.1, on the left, depicts the database design phases in general, and on the right, propagation of the same concept in the GIS system development context is found.

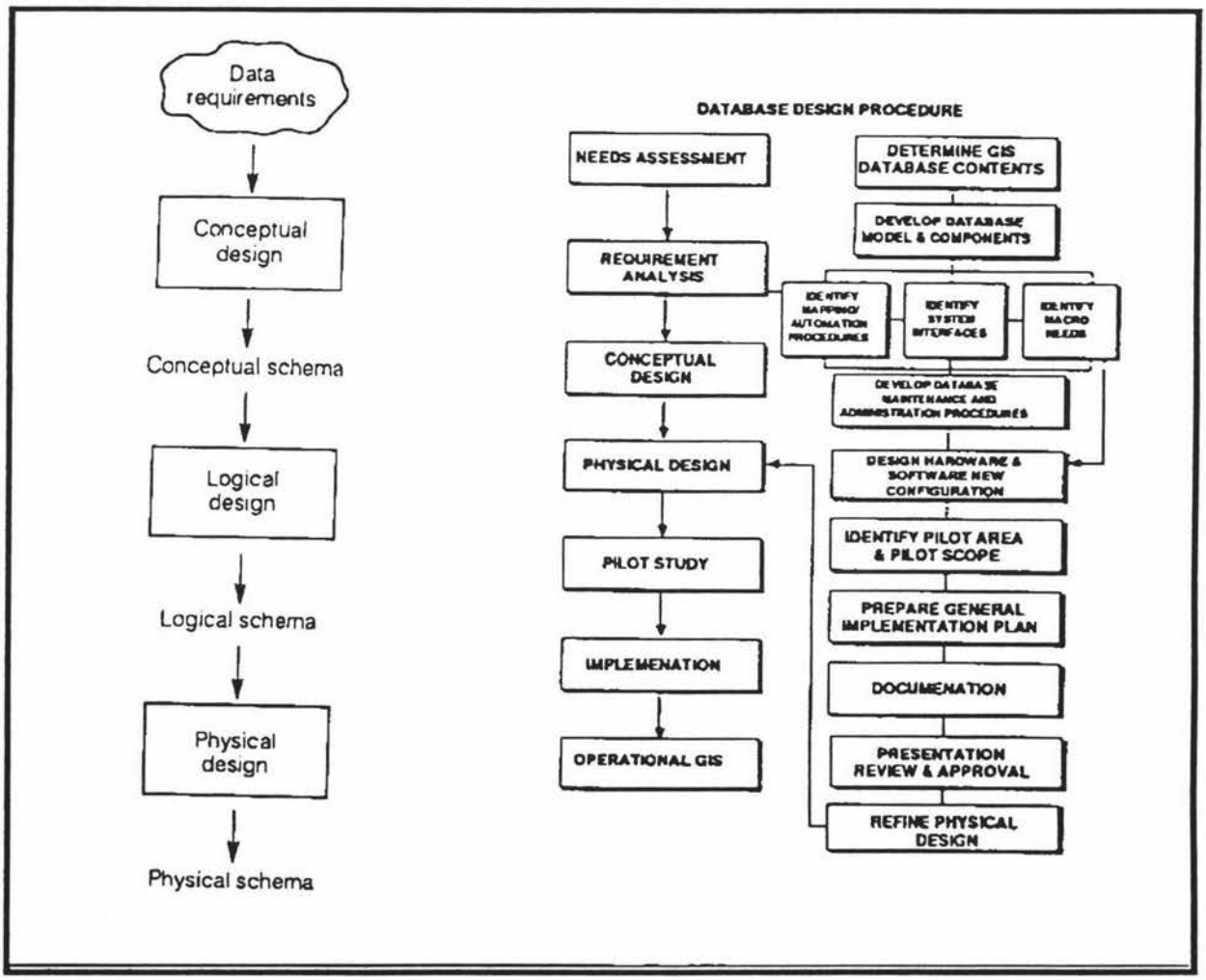


Figure 1.1 From left: General database design phases (Batini et al., 1992:7) and Overview of GIS Database Design (Chambers, 1989:2)

As shown in figure 1.1, database design is based on the requirements analysis activity, the purpose of which is to establish the description of the future database applications' data requirements, usually by using natural languages.
The first phase of database design is conceptual design. The database designer constructs a high level representation of the users' data requirements using a conceptual data model. This phase is based on the requirements analysis. The product of this first phase, the conceptual schema, eventuates by utilising the set of data modelling techniques provided by the applied conceptual data model. The conceptual schema is DBMS (database management system) independent and depicts that sub-set of reality (i.e. real world objects and relationships) about which data will be stored, and should serve as a base to the next phase, the logical database design phase (McFadden and Hoffer, 1988; Elmasri and Navathe, 1989; Date, 1991, Batini et al., 1992).

Every conceptual data modelling exercise should strive to satisfy the users' requirements while maintaining feasibility. The database designer has many responsibilities during this phase. The practitioner should carefully choose from the available conceptual data models to find the right tool to solve the problem on hand. In order to do this the designer must

- have a good understanding of the problem domain of the proposed database's applications
- keep abreast of developments in the data modelling field.

Another difficulty is that during the past decade conceptual data modelling has passed from being an "art" form into a rigorously defined engineering discipline, yet it cannot be automated. While database technology develops at a high speed, conceptual data modelling remains a human task.

It is seen in this thesis that the quality of the conceptual data modelling undertaken depends not only on the ability and hindsight gained from previous experiences of the designers, but on the tools (i.e. conceptual data model) and the training which have been provided to them.

The logical and physical data modelling phases follow the conceptual data modelling phase. Table 1.1 summarises a few, commonly confused definitions in data modelling.

•	Data modelling:	the process of designing database applications
•	Data model:	a formalism used in data modelling
•	Conceptual data model:	a DBMS independent data model (e.g. Entity relationship model)
•	Logical data model:	a DBMS dependent data model (e.g. relational model)
•	Conceptual schema:	an instance of a conceptual data model (i.e. the result of data modelling at the conceptual level)
•	Logical schema:	an instance of a logical data model (often derived from a conceptual schema)

Table 1.1 Terminology in Data Modelling (Firns, 1994b:8)

Table 1.2 stresses the differences of the various database design phases.

Dependence of	on DBMS Class	on Specific DBMS
Conceptual design	NO	NO
Logical design	YES	NO
Physical design	YES	YES

Table 1.2 Dependence of conceptual, logical, and physical design on the class of DBMS and the specific DBMS (Batini et al., 1992:8)

1.2.2 Glossary of General Terms

Database - *"a collection of persistent data used by the application systems of some given enterprise. The term "enterprise" in this definition is a generic term for any commercial, scientific or other organisation."* (Date, 1991)

Data model - *"a collection of mathematically well defined concepts that help one to consider and express the static and dynamic properties of data intensive applications. (This definition is somewhat idealistic. Most data models have evolved intuitively and have not been formally defined.) It is generally assumed that an application can be characterised by:*

- *Static properties such as objects, object properties (sometimes called attributes), and relationships amongst objects (i.e. a particular class of object properties).*

- *Dynamic properties such as operations on objects, operation properties, and relationships amongst operations (e.g. to form transactions).*

- *Integrity rules over objects (i.e. database states) and operations (i.e. state t ransitions).*

....The result of data modelling is a representation that has two components. Static properties are defined in a schema, and dynamic properties are defined as specifications for transactions, queries, and reports." (Brodie, 1984:20)

Data Base Management System (DBMS) - *"a generalised software system that is used to create, manage, and protect databases."* (McFadden and Hoffer, 1988:668)

Conceptual Data Model - *"describes entities, attributes and relationships and is independent of specific data models and database management systems."* (McFadden and Hoffer, 1988:668)

Conceptual Data modelling - *"an activity that aims to produce the global design of a database"*. (McFadden and Hoffer, 1988:668)

1.2.3 Data Modelling Paradigms

Data models are tools used for describing reality and for building database schemas. Many conceptual data models have been proposed in the literature (refer to chapter 3). The proposed data models can be classified into three general data modelling paradigms:

- entity-relationship models
- semantic data models

- object-oriented models.

These groups differ significantly in terms of diagramming conventions, but use similar building blocks on the conceptual data modelling level. These building blocks are called abstraction mechanisms[4], the means by which the designer models characteristics and properties of real world objects (Batini et al., 1992; Firns, 1994b). The primitive abstraction mechanisms are: classification, aggregation, generalisation, specialisation and grouping. These are defined and the data modelling paradigms compared in chapter 3.

1.3 Research Objectives

The principal goal of this research has been to test theoretical knowledge on a small scale - to find out if and how a conceptual data modelling method can be applied to a "real world" GIS database design problem. The following specific research objectives were set in order to achieve this goal:

(i) determine evaluation criteria which the data model must satisfy

(ii) review and evaluate the literature - identify an appropriate data model

(iii) study the selected data model

(iv) apply the selected data model.

The scope of this research has been limited mainly to the entity relationship paradigm and various extended versions of this approach, and to the object-oriented approaches for GIS database design.

The above selection of methods is not intended to be an exhaustive list of the published conceptual data modelling methods for GIS. It is acknowledged that there are developments for behaviour modelling (Frank and Egenhofer, 1988), spatiotemporal GIS modelling (Worboys, 1992; 1994), spatial knowledge representation in the field of

[4]An **abstraction** is a mental process that we use when we select some characteristics and properties of a set of objects and exclude other characteristics that are not relevant (Batini et al., 1992:15).

artificial intelligence (Smith et al., 1987b; Laurini and Thompson, 1992), the field of hypermedia (Wallin, 1990) and the field of geographical object-oriented databases (Milne et al., 1993) but this research will not consider those models.

1.4 Research Methodology

The research involved the following procedures:

(i) A review of the relevant literature.

(ii) Data requirements collection activities at the Manawatu-Wanganui Regional Council (MWRC) involving:

1. A study of the existing underground water information system.
2. An investigation of the advantages and disadvantages of the existing physical system.
3. Identifying the data needs of the applications used by the underground water sections.
4. An analysis of the data requirements and a report on the findings.

(iii) Construct a conceptual schema by applying the selected conceptual data model to the underground water database.

(iv) Draw conclusions and put forward recommendations for future research directions in this domain.

The case study research method approach has been used in preparing and conducting the data requirements activities (Yin, 1989:61-83; Zinatelli and Cavaye, 1994:17).

Data requirements collection took place at the Manawatu-Wanganui Regional Council (MWRC), Palmerston North during the period 20 - 29 June, 1994. Prior to this activity the necessary permission was obtained, a questionnaire was constructed, appointments were arranged and relevant background material was studied (refer to Appendix A, B and C).

The data collection started with an interview of the scientist responsible for the groundwater section. The interview targeted the existing database systems. (Key points: What are the general responsibilities of the groundwater section? What should be reported regularly to the management? What kind of information is given to other sections and to the public?)

After the first interview, observation of the actual work of the groundwater section was scheduled and carried out. The aim of the observation was to understand the groundwater section role in the MWRC. During the second week there was an additional interview and many informal discussions about the future plan of the section. In addition to interviews with staff and observation of the work procedures, access was gained to data files and computer facilities.

1.5 Thesis Structure and Content

This chapter has introduced the research topic, established the research objectives and methodology, and reviewed the basic definitions pertinent to the scope of this thesis.

Chapter 2 introduces and discusses all the "specialities". The definitions are given for Spatial Information Systems, Geographical Information Systems, geographical data, data structures, spatial database architectures, thematic map layers. A taxonomy of spatial data models is presented. A review of the terms relevant to an understanding of the problems to be encountered in the remaining part of this thesis is also given.

Chapter 3 reviews the literature. It is confined to a discussion and evaluation of the main stream conceptual modelling paradigms i.e. the entity relationship and the object-oriented paradigms, and their application to GIS database design. Each discussed conceptual schema is evaluated as a potential candidate for the case study.

Chapter 4 is a presentation and critique of the selected conceptual data model to be applied to the MWRC Underground Water Database case study.

Chapter 5 develops and describes the conceptual schema for the case study, based on the data requirements collection activities.

Chapter 6 contains conclusions and recommendations for future research directions in this field.

Chapter 2

What is so special about GIS?

2.1 Introduction

Chapter 1 outlined the basic concepts of conceptual data modelling for general purpose databases. In this chapter all the "specialities" pertaining to GIS databases are introduced. Definitions are provided for the fundamental terms, i.e. Spatial Information Systems, Geographical Information Systems, the nature of geographical data and geographical data structure types. These definitions apply to the remainder of this thesis. An introduction to spatial database architectures, the notion of thematic map layers and an understanding of the problem of the placement of spatial data models in the taxonomy of traditional database models are essential to comprehend the further difficulties of conceptual data modelling in the GIS field, and to an understanding of the selected conceptual data model described in chapter 4.

2.2 Spatial Information Systems and GIS

Geographical Information Systems are seen by many as special cases of information systems (Maguire, 1991; de Man, 1988; Antenucci et al., 1991; Smith et al., 1987a; Star and Estes, 1990; Aronoff, 1989; Cassettari, 1993), and include in the widest sense both manual and computer-based systems. A more precise placing of GIS in the information systems taxonomy is adopted in this thesis, i.e. GIS is a specialisation of spatial information systems (Dale and McLaughlin, 1988), this is shown in figure 2.1.

A definition of Spatial Information Systems (SIS) which is general enough to cover all types of spatial information systems is given by Laurini and Thompson (1992:22) :
"An SIS is a computerised environment whereby utility programs performing specific functions are used in an integrated environment, in which the user is shielded from the details of computer processing, to achieve some goal of research, education or decision making. The inherent form of spatial data representation and organisation must be designed to support effectively and efficiently the kind of query and analysis required by many users. The performance of computer systems is a reflection of hardware technology and software engineering, but also reflects the data structures and the quality of algorithms."

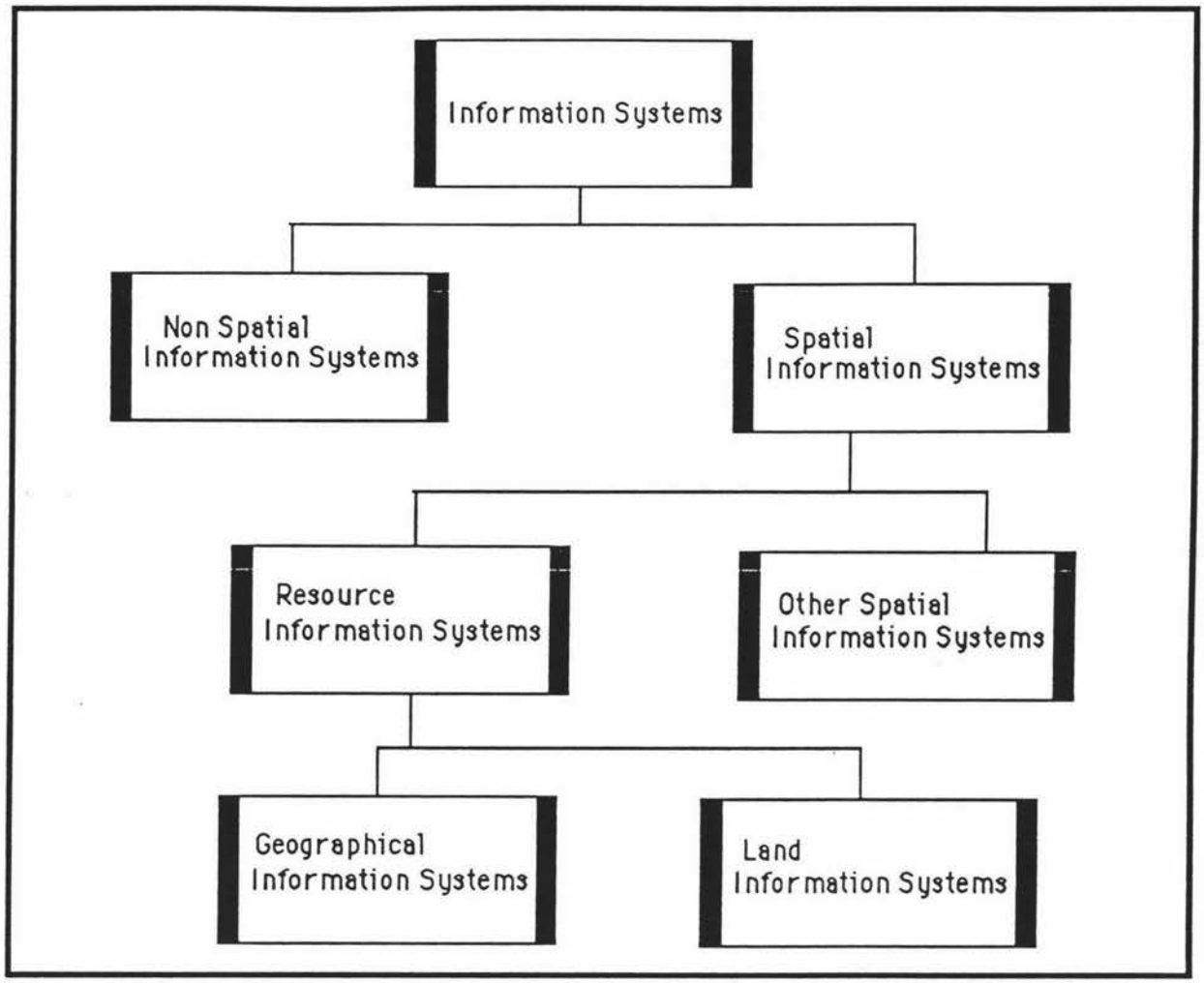


Figure 2.1 Information Systems Taxonomy (Adapted from Dale and McLaughlin, 1988:10)

In figure 2.1 GIS is placed in the taxonomy of information systems. The definition of GIS for this thesis is as follows:

"GIS is an information system comprising computer hardware, software, spatially referenced data, personnel and procedures, designed to efficiently capture, manage, manipulate, analyse, and display all forms of geographically referenced data in order to provide information for decision making." (Bekesi and Todd, 1994:203)

The main features which differentiate GIS applications from other information system applications, are the focus on spatial (geographical) data, together with specific spatial manipulative and analytical functions. This thesis is concerned only with the data needs of GIS applications. The following section provides this background.

2.3 Geographical Data, Geographical Data Structures, GIS Database Architecture and Thematic Map Layers

2.3.1 Geographical Data

Two types of data element are used to describe spatial features in a GIS database: geographical (spatial) and attribute (aspatial, textual, descriptive) data elements. Geographical data elements are the key feature of a GIS database, while attribute data elements are found in non-spatial databases (Star and Estes, 1990; Antenucci et al., 1991).

The only formal definition of **spatial information** encountered during this research is given by Firns (1994b:72):

"Spatial information is information derived by virtue of the fact that data is in some way spatially referenced (i.e. to a coordinate system.). The data defining the coordinate system and references thereto is here defined as spatial data. The role of spatial data in deriving spatial information may be that it is used in the selection, analysis or processing undertaken to produce the information, or it may be the basis for presenting the information in map form, or both. "

Euclidean dimensionality provides a basis for the four generic geographical features: points, lines, areas and surfaces. Maguire and Dangermond (1991) state *"in this scheme, points have no length dimension and are said to have a dimensionality of zero. Lines have a single length dimension and dimensionality of one. Areas have two length dimensions and dimensionality of two. Finally, surfaces have three length dimensions and give a dimensionality of three*". Further subdivision of these geographical features may be based on the properties of the associated attribute data. For example, sets of points may represent several different features such as wells, trig stations or power pylons.

The most widely used method by which attribute data is classified was proposed by Stevens (1946). His classification has four categories:

- Nominal data, so that the geographic feature can be classified.
- Ordinal data, to enable ordering of features.
- Interval data, to enable differences to be determined.
- Ratio data, so absolute values can be determined.

Table 2.1 shows the attribute data categories as they are related to the generic geographical features. The classification is provided by Maguire and Dangermond (1991).

	Point	Line	Area	Surface
Nominal	Dot	Dot	Colour class	Freely coloured
Ordinal	Ordered symbol	Ordered network	Ordered colour	Ordered colour
Internal / ratio	Graduated symbol	Flow line	Choropleth	Contour

Table 2.1 Classification of geographical data (Maguire and Dangermond, 1991:322)

This classification of geographical data has various shortcomings which are discussed in more detail by Maguire and Dangermond (1991). These include the inability to represent: networks that are more than a collection of lines, temporal change, paired feature references and problems of scale. However, the classification as shown in table 2.1 still provides a useful framework for understanding the data stored in GIS databases.

2.3.2 Geographical Data Structures

In a non-spatial database environment the term "data structure" refers to the techniques used for indexing data (Elmasri and Navathe, 1989:101), or to the way that data is stored on secondary storage devices (McFadden and Hoffer, 1988:134). In the spatial context the term has a qualifier and a special meaning attached to it. There are two fundamental, practically important geographical data structures used in representing spatial data in GIS databases: the vector and the tessellation (commonly referred to as raster) geographical data structures.
Various methods are used to implement these two "models".

In the tessellation[1] model, geographical features, i.e. real world objects, are described as values assigned to polygonal units of space in a matrix. Usually, the polygonal units are regular squares referred to as pixels, i.e. picture elements, but regular and irregular

[1]Geometrical figures that completely cover a flat surface called **tessellations** (Star and Estes, 1990:38).

triangles and hexagons have also been used. The main irregular method is the Triangulated Irregular Network (TIN) which is used to represent surfaces using triangles. Regular geographical tessellation structures include unstructured schemes such as bit maps and grids. Where the cells are organised such that they are stored in line scan order, the term raster can properly be used. At a general level the tessellation model is simple, it is readily implemented on inexpensive microcomputers and it offers a relatively quick method of developing GIS analytical operations.

In the vector model, geographical features are represented as geometric structures derived from point, line and polygon primitives, and are stored as a series of x, y or x, y, z location coordinates. Vector structuring techniques can be divided into unstructured and topological types[2]. The former can be subdivided into spaghetti, primitive instancing, and entity-by-entity structures. The spaghetti structure is so called because the geographical features are represented as a simple collection of points and lines, no spatial relationships are retained, analogous to a plate of spaghetti. Primitive instancing was developed primarily in Computer Aided Design (CAD) systems. In the database the basic elements are symbols representing buildings, roads etc., which can be moved interactively and positioned at any appropriate location on a map. The entity-by-entity structure codifies geographical features as complete units, for example as closed polygons, but no topological information is included.

Three types of topological structures are frequently identified: directional, simple and complex. Directional topological systems, such as the US Bureau of the Census DIME (Dual Independent Map Encoding) system record topology along with the direction of a line segment. Simple topological relationships are presented in systems such as POLYVERT (POLYgon conVERTer) developed at the Harvard Laboratory for Computer Graphics and Spatial Analysis. For example ESRI's ARC/INFO software system uses a fully topological model. The three major topological concepts of ARC/INFO are: connectivity (arcs connected to each other at nodes), area definition (arcs that connect to surround an area define a polygon) and contiguity (arcs have direction and left and right sides).

There has been much debate about the relative merits of both data structures, but *"In the raster-vector debate in GIS, there are no winners or losers..."* - according to Worboys et al. (1993) - *"...both views of the geographic world are natural and necessary"*. Since the tessellation data structure is area oriented, the emphasis is placed on the contents of areas rather than the boundaries between them, and so it tends to be

---

[2]**Topology** is a mathematical procedure for explicitly defining (creating and storing) **spatial relationships** between connecting or adjacent geographical features (ESRI, 1990:123).

favoured by users interested in environmental applications. The boundary oriented nature of the vector model has led to its use in a variety of socio-economic applications, especially those involving networks, coordinate geometry and high quality cartographic operations. More detailed comparison of the data structures is shown in table 2.2.

RASTER MODEL	VECTOR MODEL
Advantages: 1. It is a simple data structure. 2. Overlay operations are easily and efficiently implemented. 3. High spatial variability is efficiently represented in a raster format. 4. The raster format is more or less required for efficient manipulation and enhancement of digital images.	**Advantages:** 1. It provides a more compact data structure than the raster model. 2. It provides efficient encoding of topology, and, as a result more efficient implementation of operations that require topological information, such as network analysis. 3. The vector model is better suited to supporting graphics that closely approximate hand-drawn maps.
Disadvantages: 1. The raster data structure is less compact. Data compression techniques can often overcome this problem. 2. Topological relationships are more difficult to represent. 3. The output of graphics is less aesthetically pleasing because boundaries tend to have a blocky a blocky appearance rather than the smooth lines of hand-drawn maps. This can be overcome by using a very large number of cells, but may result in unacceptably large files.	**Disadvantages:** 1. It is a more complex data structure than a simple raster. 2. Overlay operations are more difficult to implement. 3. The representation of high spatial variability is inefficient. 4. Manipulation and enhancement of digital images cannot be effectively done in the vector domain.

Table 2.2 Comparison of Raster and Vector Data Structures (Aronoff, 1989:166)

2.3.3 GIS Database Architecture

It has been established that the data requirements of a GIS application are of two categories: spatial and descriptive data elements, giving two aspects of characteristics for one spatial feature or real world object. Spatial data are organised in a form of vector or raster model, while the descriptive data elements are organised according to the logical data model of the selected DBMS. Hence the physical architecture of GIS databases differs from the general databases, but in fact, the various GIS databases differ greatly from one another (Bracken & Webster, 1989; Healey, 1991; Worboys et al., 1993; Firns, 1994b).

Under file processing architectures data is stored in files and accessed for processing by spatial analytical software directly through the operating system, while the database approach utilises DBMS software in two different forms:

- hybrid systems or the "geo-relational model" : the spatial data is stored in files and managed by GIS software interacting with the operating system, while the descriptive data is usually stored in a standard commercial relational type DBMS such as INFO, ORACLE, or INGRES. The GIS software manages the linkage between the spatial data files and the DBMS (e.g. ARC/INFO, Environmental Systems Research Institute Ltd.).

- integrated systems or extended database management systems: also described as the spatial database management system approach. Here, the spatial and descriptive data are stored together in the same database. The standard database management system is augmented by GIS software to provide spatial analytical functions (e.g. TECHBASE software, MINEsoft, Ltd.).

2.3.4 Thematic Map Layers

On a paper map geographic information is usually organised as a set of themes, such as roads, streams, land cover types and political boundaries. They are often thought of as map layers, and each layer may actually have been plotted separately. The different types of thematic information, represented as different types of map in paper maps, are treated as different data layers in a GIS, each data layer consisting of a set of logically related geographic features and their attributes (Antenucci et al., 1991; ESRI, 1990). The features to be grouped in a single data layer are chosen for the convenience of the user. For example, the data may be organised thematically by the type of geographic features they represent. Roads and railways might be combined as a single transformation data layer and streams and lakes as a hydrology data layer.

The organisation of the data layer will also depend on the restrictions imposed by the GIS software used, it may be necessary to store point, line and area features in separate data layers. A commercially successful "vector" software, ARC/INFO is a good example of the usage of "digitised" map layers (coverages). Raster based systems, by definition, use the concept of a map layer : a cell in a tessellation can have only one value assigned to it at any given time, thus it is necessary to have multiple tessellations, distinct map layers, where two or more themes cover the same area. The "thematic map layer" is an important conceptualisation tool in the design process and in the every day

usage/access of a GIS database. This concept has a central role in a recently proposed conceptual data model (refer to chapter 4).

2.4 Taxonomy of Spatial Data Models and Levels of Abstraction

Chapter 1 defines and discusses the three phases of database design and states that each phase has a corresponding data model on three different levels of abstraction. The question posed by this section is: whether the conceptual-logical-physical data model taxonomy applies in the context of GIS database design?

Peuquet in her seminal paper "A conceptual framework and comparison of spatial data models" (1984) places the spatial data models in a "comprehensive framework" by using four levels of abstraction:

"Reality - the phenomena as they actually exist, including all aspects which may or may not be perceived by individuals,
Data Model - an abstraction of the real world which incorporates only those properties thought to be relevant to the application at hand, usually a human conceptualisation of reality;
Data Structure - a representation of the data model often expressed in terms of diagrams, lists and arrays, designed to reflect the recording of the data in computer code,
File Structure - the representation of the data in storage hardware.

The last three views of data correspond to the major steps involved in database design and implementation." (Peuquet , 1984:69)

Peuquet illustrates the four levels of abstraction, as shown in figure 2.2.

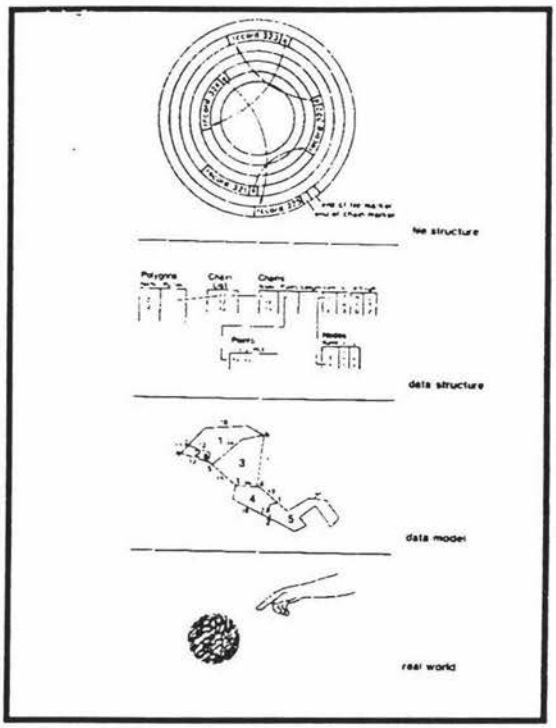


Figure 2.2 Levels of Abstraction (Peuquet, 1984:70)

After an examination of Peuquet's definitions and illustration, it is seen that Peuquet's "data model" is not a conceptual data model as defined in chapter 1, rather it is a geographical data structure, a vector model. There is confusion how to apply the traditional taxonomy of conceptual, logical and physical database models to spatial data models in Peuquet's paper.

Firns' critique of Peuquet's taxonomy concludes :

"Peuquet makes reference to vector and raster models as being at the data model level in her taxonomy. This is considered to be counter-intuitive, as a "human conceptualisation of reality" does not usually consist of points, lines and polygon, nor does it consist of a grid of regular shaped and sized 'pixels' of an arbitrarily defined resolution...such models lie somewhere between conceptual models and logical models, and probably closer to logical models." (Firns, 1994b:78)

Firns observes that the three level taxonomy of data models is not readily applicable in a spatial context, because there is an additional level of abstraction between the conceptual and the logical data model levels. The proposed taxonomy of spatial data models and levels of abstraction is shown in table 2.3, *"where each level of model corresponds to a different level of abstraction in database design"* (Firns, 1994b:81).

Firns also states that there is no direct correspondence between the traditional database design phases and the levels of abstraction associated with spatial database design as presented in table 2.3. The traditional database design phases would be applicable in a spatial database design context if the notion of spatial representation was present at the conceptual database design phase. The reasoning behind this suggestion is explained by Firns as follows. The varied spatial representation models are different in their semantic expressiveness. If the tessellation model is to be adopted, there is little practical use to construct semantically rich conceptual schemas, because that model is devoid of semantics.

Model Level	**Abstraction Level**
Reality	the real world as it is perceived
Conceptual data model	a model of real world semantics
Spatial representation model	roughly corresponding to Peuquet's data models - *the general concepts of raster and vector models [as seen by Firns]*[3]*
Spatial data model (logical level)	corresponding to Peuquet's data structures - *various methods to implement the 'raster' and 'vector' e.g. TIN or topological model [as seen by Firns]**
Spatial data structure (physical level)	corresponding to Peuquet's file structures

Table 2.3 A Taxonomy of Spatial Data Models and Levels of Abstraction (Adapted from Firns, 1994b:81)

The first level in the taxonomy is *"Reality - the real world as it is perceived"* - interpreted in this thesis as - by the database designer or database administrator. Reality is defined as the phenomena it is fundamentally important to model in information systems and about which canonical or raw data is collected. Facts which are derivable from the basic collection of data about the phenomena, data used to represent the phenomena according to a particular representation model (e.g. line segment for river) and generic representations of geometric features (e.g. point, line, area) are not part of this definition. Maps, even though they may be considered to have an independent existence as graphical objects, are also not part of the current definition.

---

[3]The explanatory sentences, marked with "*" in italics, has been added by using Firns' ideas (Firns, 1994b:81-82).

2.5 Chapter Summary

This chapter reviewed the basic specialities of data stored in GIS databases, i.e. the two types of characteristics of one geographical feature to be represented in a spatial database, and the confusion of relating the non-spatial data model taxonomy and the spatial data models. Both these issues constitute extra hardship for a database designer when adjusting traditional database design techniques in this special problem domain.

Chapter 3

Literature Review

3.1 Introduction

This chapter reviews a group of published papers, selected from proponents of the entity-relationship and object-oriented data modelling paradigms. The two data modelling paradigms are chosen for different reasons. The ER approach is widely used in conceptual database design, especially in non-spatial environments. Its benefits are *"providing a high level of abstraction, displaying database constraints and interrelationships, and providing easy-to-understand notation"* (Czejdo et al., 1990:26). The second, the object-oriented paradigm cannot be overlooked as it is "hot" in programming languages and data management environments. "My Cat is Object-Oriented" - declares King (1989), illustrating somewhat sarcastically the point that "object-oriented" things are selling well these days, though the real potential of this paradigm is yet to be realised (Batra et al., 1992:398).

Different methods are used in the surveyed literature. In reviewing the ER approach, first the original data models are introduced (i.e. Chen, 1976; Elmasri et al., 1985; Czejdo et al., 1990), then the application of these models in the SIS context are shown. Each presented spatial conceptual schema is evaluated from the perspective of this thesis. The lack of a generally accepted object-oriented data model[1] in the database domain leads to the second method. Detailed critique of one paper (Worboys et al., 1990) is used to study concepts pertaining to the object-oriented paradigm and to discuss a semantic data model. Finally, a comparison of the main data modelling paradigms is included.

This thesis sets out to find a conceptual data model applicable to GIS database design. Some authors (Brodie, 1984; Bedard and Paquette, 1989; Firns, 1994a) recognise the need for a specific conceptual data model to produce suitable conceptual schemas for databases in specific problem domains. This thesis takes the view that the application of any general or specialised data model must serve the aim of the database design. For this reason the objectives of conceptual data modelling are chosen to evaluate data models encountered throughout the research documented in this thesis. The objectives of

[1]There are positive developments recently - an issue considered later in this chapter (Cattell, 1994).

conceptual data modelling are to describe reality and construct the base of the logical database design (Batini et al., 1992; Elmasri et al. 1985, etc.). A definition is given by Firns (1990a:13) :

"...there are two major objectives to meet when developing data models:

(a) to adequately and accurately represent in an understandable manner, real world phenomena and relationships that may exist between them.

(b) to develop a model with sufficient rigour to form the basis for a database structure in which specific instances of the real world phenomena may be represented in the form of data values."

3.2 ER-based approaches

3.2.1 The Entity - Relationship Data Model

The entity-relationship (ER) data model is proposed by Peter Chen, in his paper "The Entity-Relationship Model - Toward a Unified View of Data", published in 1976.

In the introduction the author justifies the necessity of a new data model by stating that the three major data models[2] of his time may achieve data independence, but loose important semantic information about the real world. Since then, Chen's opinion on the shortcomings of record-based data models is well supported by Kent's seminal paper (1979) among others (e.g. Schmith and Swenson, 1975 [quoted by Batra et al., 1990:395]; Smith and Smith, 1977a, 1977b; Hammer and McLeod, 1981; Tsichritzis and Lochovsky, 1982).

Chen's data model strives to create a framework from which the three above mentioned data models may be derived. The main characteristic of the ER model is that it is a conceptual data model and *"adopts the more natural view of the real world"* (Chen, 1976:9) Instances of the ER model are developed to represent the sub-set of the reality about which data is to be stored in the database, without considering the physical implementation details of that storage.

---

[2]Network model (Bachman, 1969; CODASYL, 1971), relational model (Codd, 1971) and the entity set model (Senko et al., 1973).

The paper consists of four parts: the first part introduces the entity-relationship model, the second describes the semantic information of the model and its implication for data description and data manipulation (set operations), the third part contains the diagrammatic technique and the last part of this much referenced paper analyses the derivation of the logical models. For the purpose of this thesis the second and the fourth parts of the paper do not hold interest.

Entity sets, relationship sets and attributes

The basic blocks of the model are entity sets and relationship sets. *"The database of an enterprise contains relevant information concerning entities and relationships in which the enterprise is interested."* (Chen, 1976:11) Definitions of these basic elements are given in the paper:

"An entity is a "thing" which can be distinctly identified...Entities are classified into different entity sets." (Chen, 1976:10-11)

"A relationship is an association among entities...A relationship set, R_i *, is a mathematical relation among n entities, each taken from an entity set:*
$\{ [e_1, e_2, ..., e_n] / e_1 \varepsilon E_1, e_2 \varepsilon E_1, ..., e_n \varepsilon E_n \}$,
and each tuple of entities, $[e_1, e_2, ..., e_n]$ *is a relationship.."* (Chen, 1976:12)

E_1, E_2 etc. in the above definition may not be distinct. An example of an entity set is PERSON and "marriage" is a relationship set between two entity occurrences of the PERSON entity set.

"The role of an entity set in a relationship set is the function that it performs in the relationship." (Chen, 1976:12) e.g. "husband" and "wife" are roles that a PERSON instance can play in the marriage relationship occurrence.

Entity sets and relationship sets are described by common properties, i.e. a set of attributes. Attributes are prominent components of the ER model. The value of an attribute represents the observed or measured information about an entity instance or a relationship instance and these values are taken from pre-defined value sets. An attribute is formally defined as:
"a function which maps from an entity set or a relationship set into a value set or a Cartesian product of value sets:

$f: E_i$ *or* $R_i \rightarrow V_i$ *or* $V_{i1} \times V_{i2} \times ... \times V_{in}$." (Chen, 1976:12)

More than one attribute may map from an entity set into the same value set. For example, NAME and ALTERNATIVE-NAME map from EMPLOYEE into value sets FIRST-NAME and LAST-NAME.

The concept of a primary key (Chen, 1976:14) pertains to attributes. This concept was introduced by Codd (1970), and since Chen's paper it has been extended, as summarised by Kennedy (1993: 57):

"A requirement of the relational model, (but true of others too,) is that, for each entity set, one attribute must be named as the 'entity identifier' or primary key of the table, that is, a data-item whose value is unique for each record occurrence in that table. The identifier may be a natural attribute, such as NAME or COLOUR, but often it is an internally allocated (Finkelstein, 1989: 141) or surrogate (Dittrich, 1987:135) attribute, whose uniqueness is guaranteed by the system, and which is then used as a primary key in the corresponding database table. Though the surrogate key does not correspond to any real world property, it frequently becomes a surrogate for the physical entity itself, as in the case of a product number, bank account number, or student identification number."

The concepts discussed in the above excerpt will be put to use later in this chapter.

The diagrammatic technique of the ER model, as proposed by Chen is illustrated in figure 3.1. Entity sets are represented by rectangles and relationship sets by diamonds.

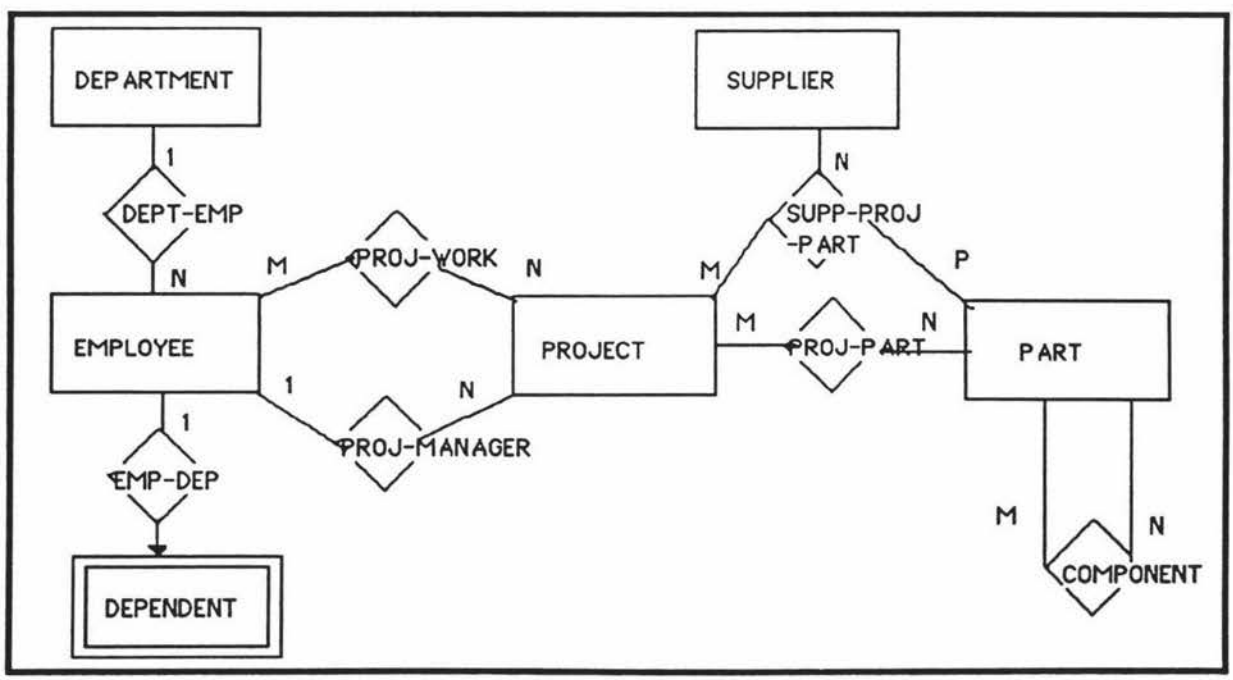


Figure 3.1 An Entity-Relationship diagram (Chen, 1976:19)

Further description of the ER model is provided in the next two tables. Table 3.1 shows four classic entity/relationship types identified in Chen's model and table 3.2 summarises the properties of relationships in the ER model.

Entity / Relationship	Characteristic
Regular entity e.g. EMPLOYEE	Can exist without being related
Weak entity e.g. DEPENDENT	Depends on other entities for existence
Regular relationship e.g. PROJECT-EMPLOYEE	A 'relationship' which can exist on its own
Weak relationship e.g. EMPLOYEE-DEPENDENT	Not allowed to have dependent entities

Table 3.1 Classic Entity/Relationship Types (Adapted from Kennedy, 1993:66)

Characteristics of relationship sets	Examples from figure 3.1
A relationship set may be defined for more than two entity sets.	e.g. the SUPPLIER - PROJECT - PART relationship set
A relationship set may be defined for only one entity set.	e.g. the COMPONENT relationship set is defined for one entity set, PART
There may be more than one relationship set defined for any given entity set.	e.g. the relationship sets PROJECT - WORKER and PROJECT - MANAGER
Information about the number of entities from each entity set which are allowed in a relationship set* are indicated by specifying "1", "M", and "N" in the diagram	e.g. the DEPARTMENT - EMPLOYEE relationship set is a 1:n mapping, the relationship set PROJECT - WORKER is an M:N mapping
Existence dependency of one entity type on another.	e.g. the arrow in the relationship set EMPLOYEE - DEPENDENT indicates that an instance of the entity set DEPENDENT depends on a corresponding instance of the entity set EMPLOYEE.

Table 3.2 Properties of Relationships (Adapted from Chen, 1976:20)

*Generally termed as relationship cardinality. It is appropriate here to define the two widely accepted notions of constraint on relationship sets. The first is the **cardinality constraint,** it specifies the number of relationship instances that a specific entity can take part in; and the second is the **participation constraint**, that specifies whether the existence of an entity depends on its being related to another entity via a relationship set, in other words, whether the participation of an entity instance in the relationship is optional or mandatory. (Indicated by "O" and "M" symbols, which are extensions to Chen's original notation.)

Many published papers apply the original ER model[3] (e.g. Lipeck and Neumann, 1987; Burdock, 1987; Calkins and Marble, 1987; Goh, 1988; Feuchtwanger and Poiker,

[3]The first attempt at using the ER diagram and extensions to it for spatial database design is provided by Nyerges (Nyerges, 1980, quoted by Nyerges, 1989:157).

1987). Two of these papers are reviewed in this thesis (Wang and Newkirk, 1988; Laurini, 1991).

3.2.2 Wang and Newkirk's ER Model for GIS Development

"An Entity-Relationship Model For Geographical Information System Development" is the title of Wang and Newkirk's paper, published in 1988.

The paper sums up the state of knowledge in data modelling. Applying the ER techniques to a "Geographical Information System [sic]" is a groundbreaking aspect of the authors' work. These characteristics are the main strength of the paper.

The major problem is the authors' vagueness on definitions. The paper does not differentiate the various levels of data models or between a data model and an instance of a data model. The paper gives the following definition of GIS:

"GISs are designed to handle ,represent or describe spatial entities. *An important feature of a GIS is the frequent creation and manipulation of spatial thematic map displays. This requires that a GIS should be able to handle the spatial representations as well as the non-spatial descriptions."* (Wang and Newkirk, 1988:163)

The relevance of the definition becomes clear when the authors list the ingredients of a GIS data model. They are: the *"generic types of the spatial entities, their attributes, and relationships...a GIS data model should include the spatial representations and display features for graphics output."* (Wang and Newkirk, 1988:164)

Figure 3.2 shows the ER schema presented in the paper. It is seen that the authors do not exclusively model reality. Wang and Newkirk model "spatial entities" (e.g. City) together with their "spatial representation" (e.g. Polygon), using the original relationship sets to depict spatial relationship sets (e.g. City_Poly). The ER schema also contains attributes to specify the physical appearance of the output (i.e. colour of a map), which are not considered for inclusion at the conceptual level.

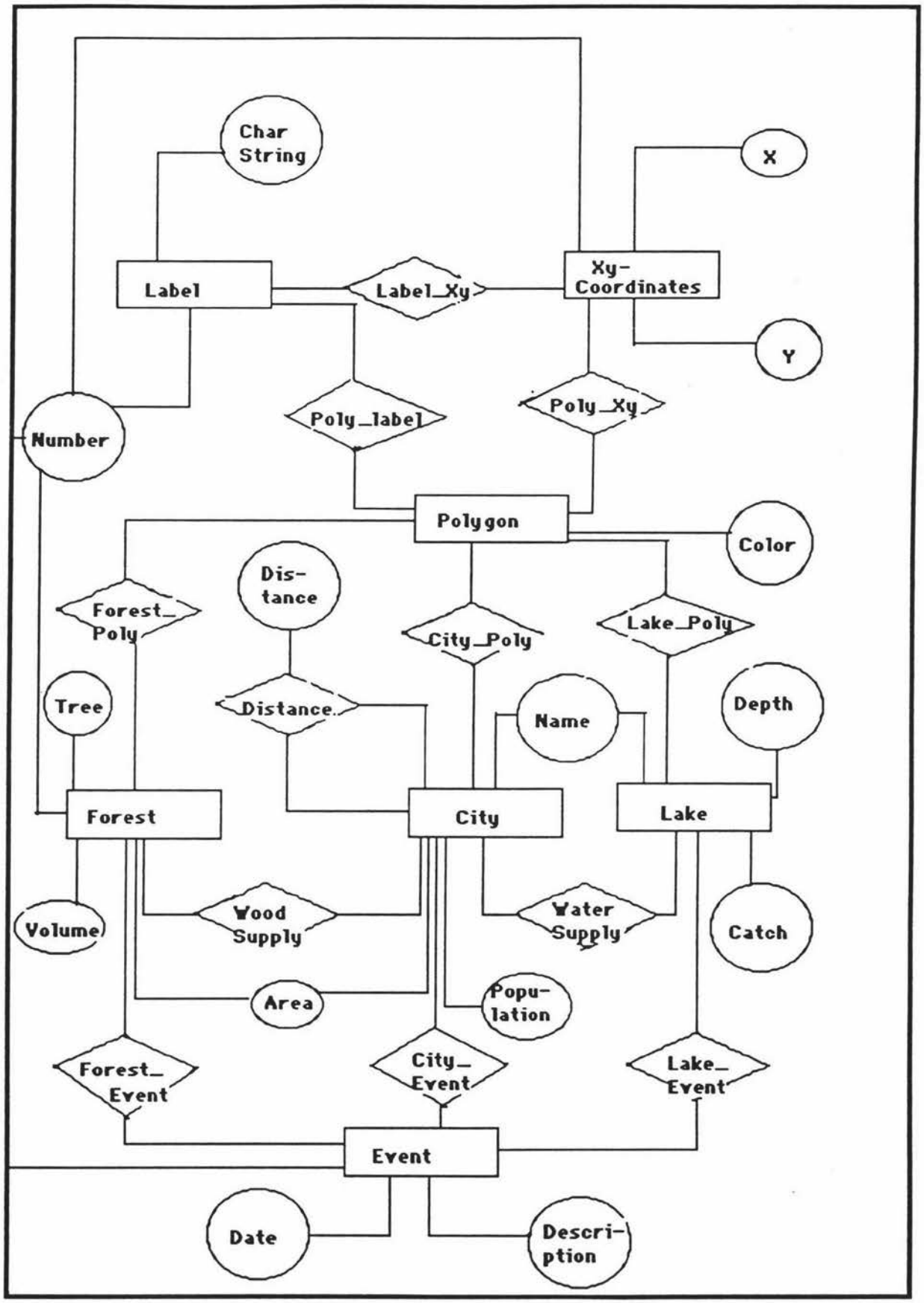


Figure 3.2 ER Model for GIS (Wang and Newkirk, 1988:169[4])

[4]This figure has been reproduced with the help of Firns' paper (1992).

The main purpose of this thesis is to find a conceptual data model to apply to the case study. In order to decide if the data modelling technique presented by Wang and Newkirk is potentially successful, it must be evaluated against the chosen evaluation criteria.
The objectives of the conceptual data modelling activity are briefly:

" • *to model reality and*
• *to form the basis for database schema design* " (Firns, 1994b:43)

Reality is defined in this thesis as the phenomena it is fundamentally important to model in information systems and about which canonical or raw data is collected (refer to chapter 2.4). This thesis recognises that the different levels of perception and separate models of real world phenomena and their representations are essential in a spatial context.

If Wang and Newkirk's schema is measured against the first criterion, it goes too far. Reality and its representation are modelled in the same schema, they fail to separate reality from its representation. This is a problem with Wang and Newkirk's modelling approach rather than with the ER approach itself. The conclusion must be that the paper does not present an appropriate technique for the case study.

3.2.3 Laurini's ER Model of Urban Data

Laurini published his "Introduction to ER Modelling for Urban Management" paper in 1991. The main strength of Laurini's work is the suggestion of an extended ER methodology specifically applicable to Spatial Information Systems design. The weakness of the paper is that the proposed extensions do not help to model reality.

Laurini proposes two extensions to the original ER data model:

- incorporation of the synthesis of multiple geometric representations for the same urban object, and
- incorporation of an extensional/intensional aspect of spatial data[5].

Multiple representation of spatial data is explained as *"...a feature of spatial information is that an object can be defined or represented by several methods"* (Laurini, 1991:480). For example a road can be represented as a graph or a set of geodetic points or a set of 2D segments (parcel boundaries) or a waiting list (traffic simulation) etc..

[5]The same data model reappears in Laurini and Thompson 's textbook (1992:362-666).

The extensional/intensional aspect of spatial data is *"that rules are necessary to derive the extensive expression"* (Laurini, 1991:475).

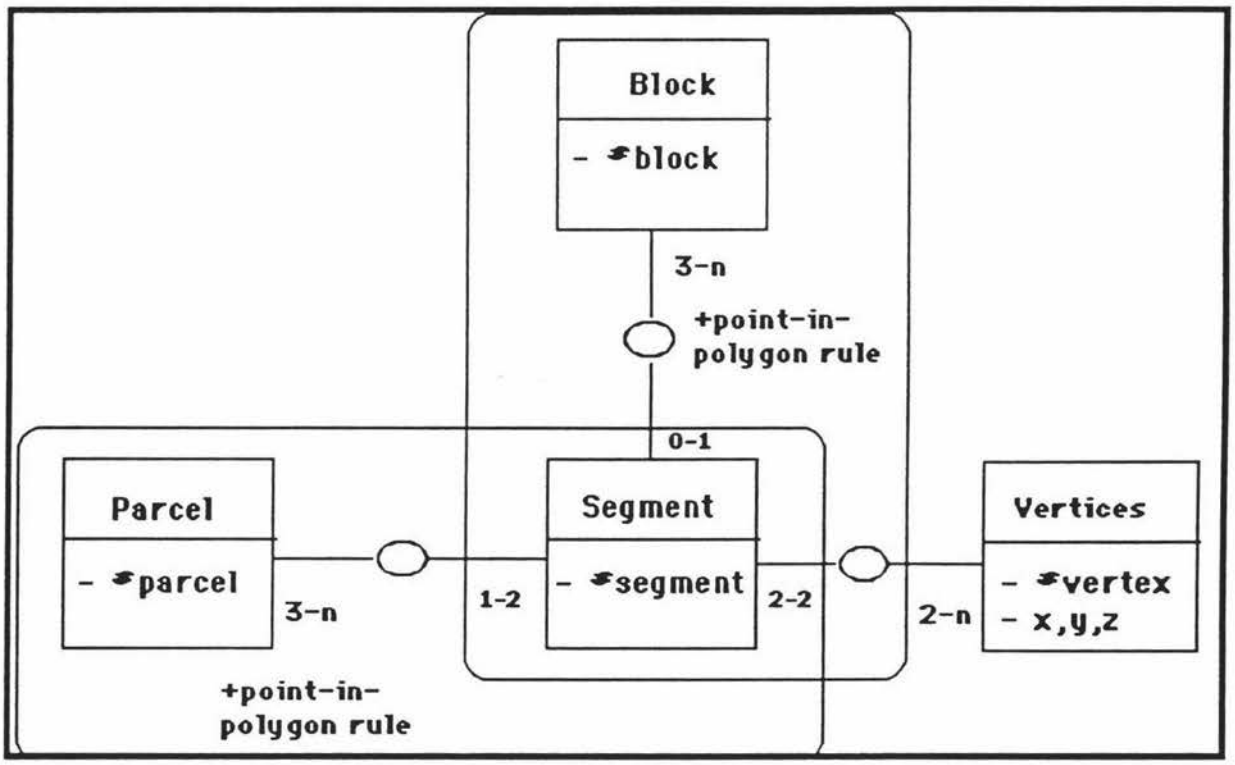


Figure 3.3 Conceptual model for parcels and blocks (Laurini, 1991:481)

Figure 3.3 shows a modified ER model for parcel and blocks. The diagram reads: Parcel is an entity set, with an attribute #parcel. The relationship set between the Parcel and Segment entity sets is denoted by a circle or ellipse, the minimum and maximum cardinality are denoted by variables placed alongside the entity sets, i.e. a Parcel has at least three up to n boundary segments and the same Segment delimits one or two Parcels. The two big rectangles indicate that figure 3.3 is the synthesis of two different external models. The rules (e.g. +point in polygon rules) placed under/next to the relationship set lines are a guide to find all the points belonging to a specific parcel/block/segment.

Laurini's modelling approach does not differentiate between real world objects and their spatial representation, Parcels (real world objects) are modelled together with Vertices (spatial representations). If Laurini's model is evaluated against the objectives of conceptual data modelling (to model reality and to form the basis for database schema design; Firns, 1994b:43) this approach does not succeed. Thus the technique offered by Laurini is not appropriate to the case study.

Following the publication of Chen's data model (1976) there have been many extensions proposed for ironing out its shortcomings. (e.g. Chen, 1979, 1981, 1985; March, 1988; Spaccapietra, 1987 [quoted by Firns, 1990:14]; Scheuermann et al. (1979); Dos Santos et al. (1979) [quoted by Elmasri et al. (1986:451] and Teorey et al., 1986). These semantically richer models are generally called extended entity relationship models. Two of these models are discussed in this thesis (Elmasri et al., 1985; Czejdo et al., 1990).

3.2.4 The Entity-Category-Relationship Data Model

"The category concept: An extension to the entity-relationship model" is the title of Elmasri's et al. paper, published in 1985. The authors propose an enhanced version of the ER data model, called the Entity-Category-Relationship (ECR) data model[6].

The paper states that Chen's ER model is not sufficient to represent some important data semantics i.e. the notion of subclasses (ISA hierarchies) and of super-classes (generalisation) are not directly supported by the ER model. Another concept is the grouping of entities which do not belong to the same entity set, participating in the same relationship.

"Briefly, the ECR model, like the ER model, views the world as consisting of entities and relationships among entities. Entities have attributes which provide information. Entities are classified into entity types according to their basic attributes. Furthermore, in the ECR model, entities are grouped into categories according to the roles which they may play in relationships. Thus, relationships are mathematical relations over categories of entities." (Elmasri et al., 1985:78)

The extensions to the original ER model are:

Entities and relationships

Similar to Chen's model, an entity is something which exists in the real world and has attributes to describe it. An extension to Chen's model is that an entity has basic attributes and acquired attributes. The acquired attributes are not fundamental to the entity set, but are the result of a relationship set involving the entity set. (E.g. PERSON has basic attribute hair-colour and an acquired attribute, employee-number, which is the result of the "employment" relationship between the PERSON and COMPANY entity sets.)

---

[6]It is interesting to note that Elmasri later co-authored a different data model, called the enhanced-ER or EER model, for teaching purposes (Elmasri and Navathe, 1989:409).

Entity types and categories

Entities which have similar basic attributes are classified into entity types. Entity types are disjoint. Entities are also classified into categories:

"A category is a set of entities from one or several entity types, that play a role in a relationship, that generalise other entities or that specialise other entities." (Elmasri et al., 1985:79)

Categories are not necessarily disjoint. For example: CAR, TRUCK, PERSON and CORPORATION are entity types. CAR and TRUCK entity types may be categorised as VEHICLE in an "owner-vehicle" relationship. PERSON and CORPORATION entity types may be categorised as OWNER in the same relationship.
Categories are also used to represent subsets (ISA-categories). An example is the three categories of the EMPLOYEE entity type: FULLTIME-EMPLOYEE, SCIENTIST and TECHNICIAN.

Relationships

The ECR model takes a more complex approach to specifying relationship participation and cardinality than the original ER model. A mandatory relationship is termed total and an optional relationship is termed partial. Two further types of participation are identified by the new model: functional and specific participation, these correspond to 1:1 relationships and dependent entities in the original ER model. The ECR model also specifies minimum and maximum cardinality for each entity set (Elmasri et al., 1985:80). An example of partial/total participation of categories in a relationship are the OWNER and VEHICLE categories in the "ownership" relationship. Since not all PERSON and CORPORATION entity type instances own vehicles, their participation is partial. On the other hand, the participation of the VEHICLE category is total, a vehicle must have an owner.

"The total participation of a category in a relationship implies an existence dependency of entities within that category upon related entities within other participating categories."(Elmasri et al., 1985:80)

However the participation of the VEHICLE category in the "ownership" relationship is not specific, since vehicles may be reassigned to new owners. An example of a specific relationship is given in the paper i.e. between the EMPLOYEE and DEPENDENT categories the "supports" relationship is specific. (In practice, deletion of the supporting EMPLOYEE from the database would imply deletion of the DEPENDENT.)

Cardinality in the original ER model is the property of the relationship itself (refer to table 3.2), while in the ECR model, it is the property of each entity set participating in a relationship.

Attributes

Attributes (basic or acquired) are associated by entity types, categories or relationships.

"Each attribute is defined on a value set, from which the values of the attribute for the entities or relationship instances are taken...An attribute a, of an entity type T, a category C, or relationship R, defined on values set V, is a function with domain T, C, or R, and range P(V) is the power set of V:

a:T -> P(V) *or a:C -> P(V)* *or a:R -> P(V).*

This definition allows for a direct representation of multi-valued attributes." (Elmasri et al, 1985:81.)

Primary keys

"...entities stand for themselves ...This approach is similar to the concept of surrogates." (Elmasri et al, 1985:77.)

The notion of a surrogate key is discussed in section 3.2.1.

Diagramming Conventions

Figure 3.3 shows a simple ECR diagram. An entity type and a relationship set are represented as in Chen's diagrams. The hexagonal box, connected to the entity types by directed lines represents a category. If the category is identical to an entity type a hexagon is drawn around the original entity rectangle box.

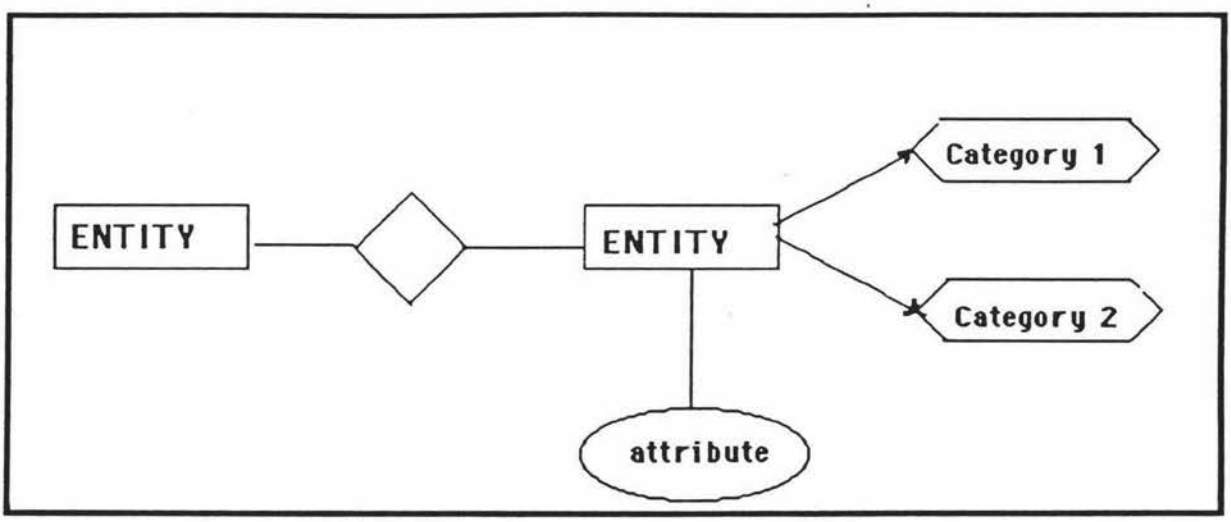


Figure 3.4 ECR Basics (Armstrong and Densham, 1990:9)

Table 3.3 lists the new terms of the Entity-Category-Relationship data model.

ECR - SUMMARY OF NEW TERMS
• Entity type
• Category (ISA)
• Category (generalisation)
• Functional relationship (1 : 1 ; 1 : N)
• Non - functional relationship (M : N)
• Multi - valued attribute
• Entity "identifier"

Table 3.3 New terms of ECR (Adapted from Elmasri et al., 1985:89)

3.2.5 ECR for a Spatial Decision Support System

Armstrong and Densham's "Database organisation strategies for spatial decision support system" paper, published in 1990, uses the Entity-Category-Relationship data model to put forward a conceptual database design methodology for spatial decision support systems (SDSS).

The paper covers a wide range of topics. It reviews the literature of decision support systems, defines the term SDSS, proposes a conceptual database design methodology, introduces the available logical database models for SDSS, specifies steps for deriving logical database structure from the conceptual model, and then suggests a particular logical database model for developing databases for SDSS applications. This thesis is concerned with section 3, "Rationale for using DBMS technology", and section 4, the introduction of a "conceptual database design methodology".

Rationale given by the authors:

"The way in which information is organised in any computer system is a critical factor in its success or failure, and the database is therefore the foundation upon which the SDSS operates....Data in computer systems can be viewed at different levels of abstraction.. In this paper, a comparatively simpler three-level strategy is used derived from the ANSI/SPARC framework (Tsichritzis and Klug, 1978), and related to that of Peuquet (1984) and that used by Bestougeff (1984) for database design." (Armstrong and Densham, 1990:7)

It is not clear how Armstrong and Densham use Peuquet's four levels of abstraction in their work.

"A common problem in database design is the need to accommodate varying views of existing users in a final database. In the case of designing an SDSS, at least two categories of views must be considered: the cartographic and the analytic. Fortunately, these categories are complementary, making the process of integration relatively straightforward." (Armstrong and Densham, 1990:8)

Figures 3.4, 3.5 and 3.6 show the cartographic view, the spatial analytical view, and the integrated SDSS view respectively.

In figure 3.4, the cartographic view shows the topological chain model representing features, used in digital cartographic and GIS applications. The model reduces the entities to nodes, chains and polygons, each built from a set of points (coordinates). The authors' explanation of the recursive relationship (Member_of) is:

"The recursive relationship (Member_Of) permits hierarchical aggregation of polygonal entities (e.g. aggregating counties into states)." (Armstrong and Densham, 1990:10)

Usage of the word "aggregation" by a general dictionary definition meaning, and not as the term is used in the data modelling context, is confusing (Firns, 1992:24).

The spatial analytical view (figure 3.5) has many similarities to the cartographic view, the only difference is the presence of attributes, *"...because of the prominence of the information used in analyses. In figure 3.5, this information is represented by an entity set. This is done because the great many ways in which the attribute data can be maintained (e.g. sorted data, sorted by spatial units, and grouped into levels of measurement)"* (Armstrong and Densham, 1990:10).

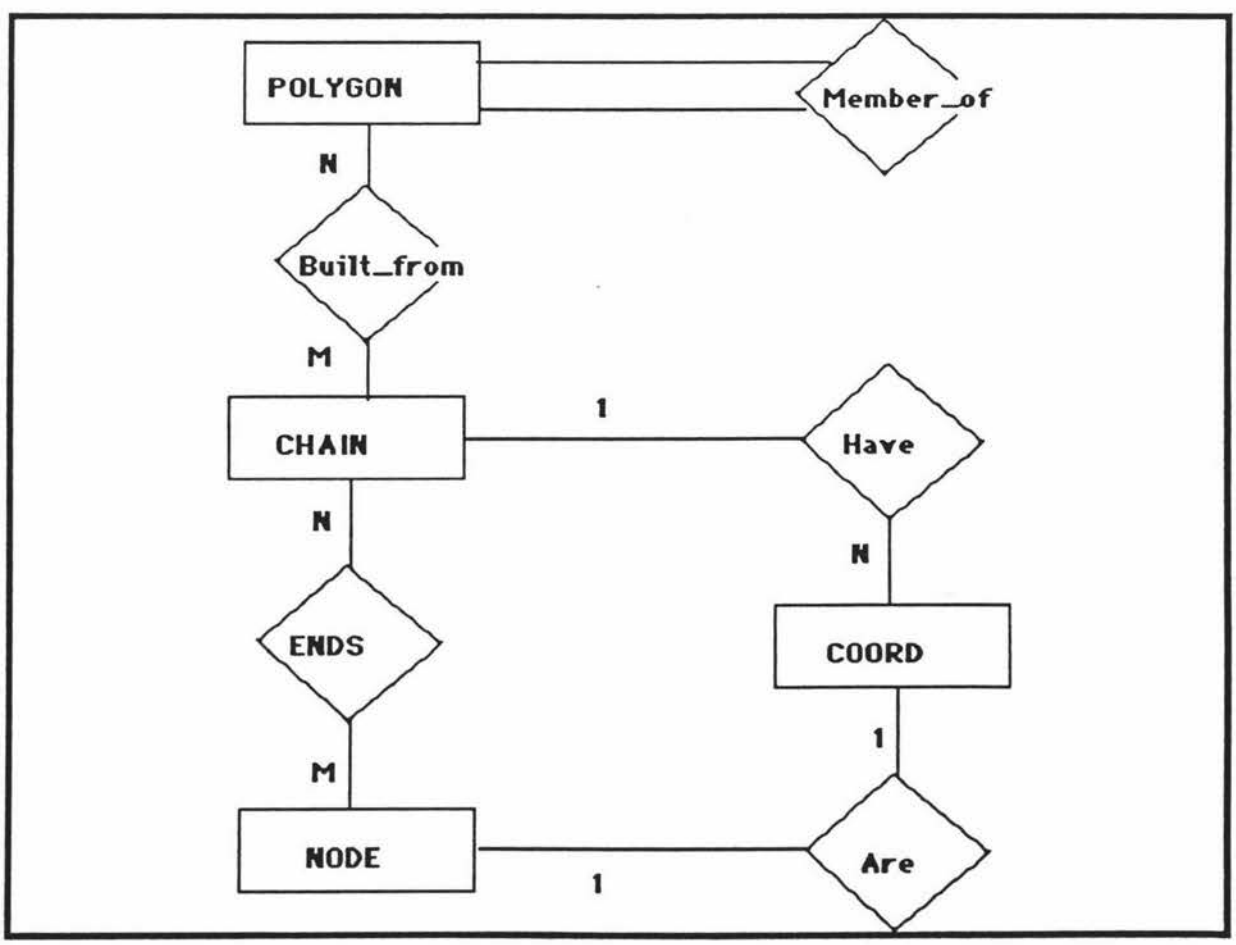


Figure 3.4 The ECR cartographic view (Armstrong and Densham, 1990:9)

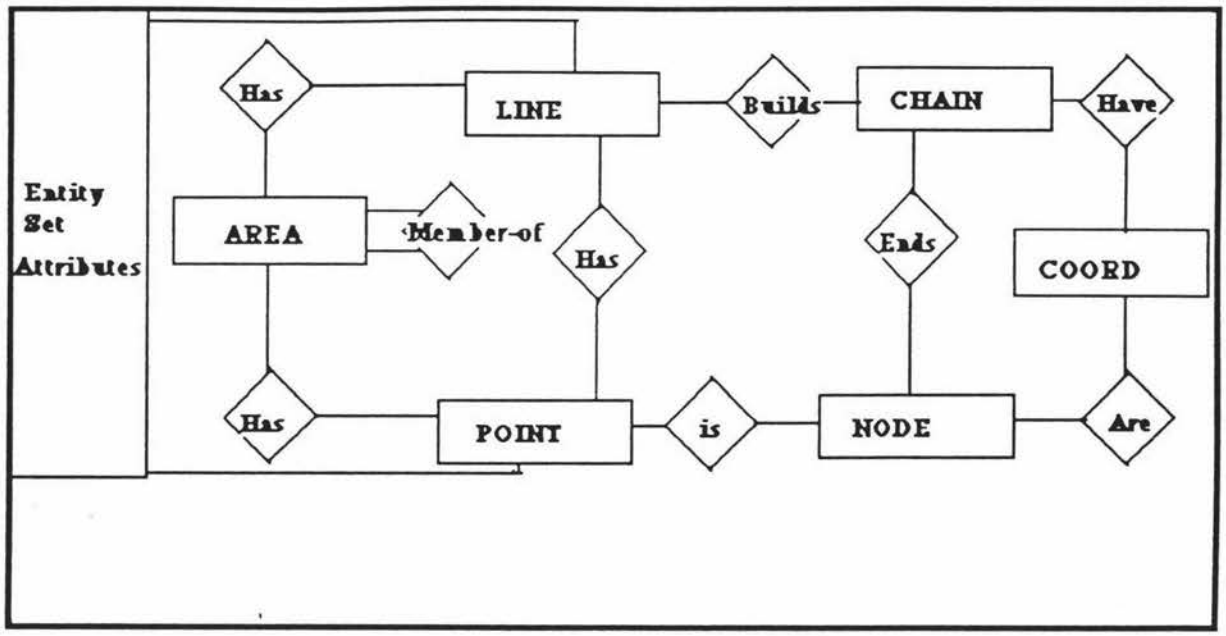


Figure 3.5 The ECR spatial analytical view (Armstrong and Densham, 1990:10)

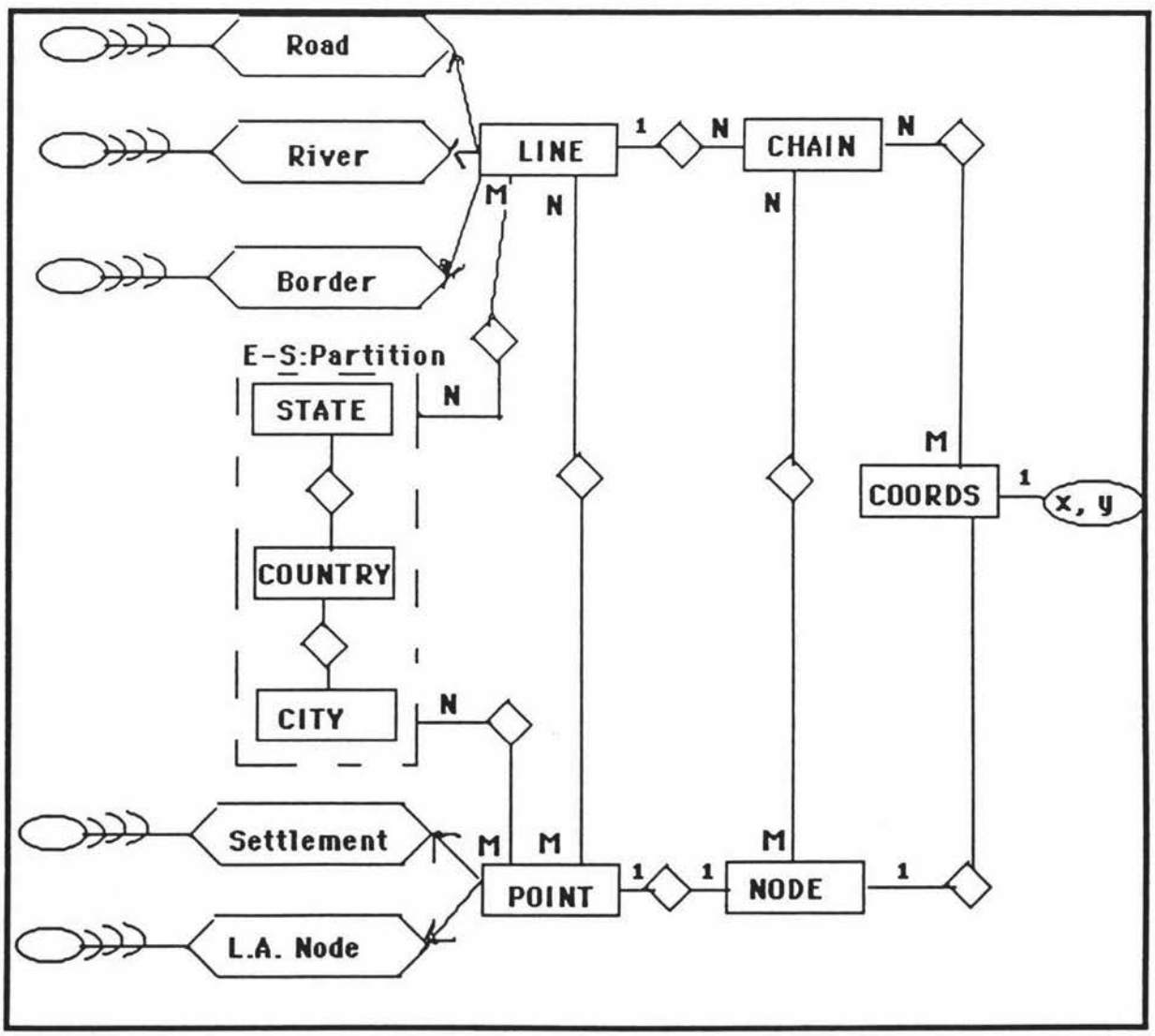


Figure 3.6 The integrated SDSS view (Armstrong and Densham, 1990:11)

The authors' account of figure 3.6 is that the SDSS view combines features from the two previous views and extend it with other types of operations. The cartographic component is represented by the chains, nodes and coordinates on the right side of the figure, the analytical component is built from nodes, lines and areas. They use a unique notation, 'E-S Partition', to depict areas.

The authors recognise that *"... a line is a spatial object used to represent an entity."* (Armstrong and Densham, 1990:8), but still integrate the real world entities with their spatial representation. The proposed modelling technique does not exclusively model reality, and the technique does not comply with the objective of this thesis. Therefore the ER schema presented by Armstrong and Densham is not appropriate for the case study.

3.2.6 The Extended Conceptual Entity-Relationship Model

The Extended Conceptual Entity-Relationship Model is proposed by Czejdo et al. in their paper "A Graphical Data Manipulation Language for an Extended Entity-Relationship Model" paper, published in 1990.

The paper states that using graphical interfaces to formulate queries and to execute update operations provides for an easier interaction with database systems. A special type of graphical interface is suggested by the authors - the conceptual schema diagram of a database. The query formulation on the displayed schema diagram lends itself to a convenient "point and click" interface for a database system.

The authors argue that *"An ER schema diagram captures many of the constraints and semantics of a database, but some important semantic concepts cannot be expressed using the basic ER model"* (Czejdo et al, 1990:26).
The paper formally defines the ECER model and uses the university database ECER schema to illustrate the concepts of the proposed data model. The schema diagram is shown in figure 3.7.

Entity sets
"Each entity set in the schema represents a set of entities in the database that are of the same type and share some common properties or roles." (Czejdo et. al, 1990:28)

In figure 3.7, entity sets are represented by rectangular boxes. An example of an entity set is PERSON, which represents all persons known to the database. An entity

occurrence is identified by a surrogate-key value and the identifier is not shown on the diagram.

Relationship sets

"...represent a set of relationship instances, where each instance relates one entity from each of the participating entity sets." (Czejdo et al., 1990:29)

A relationship set in an ECER diagram is represented by a diamond-shaped box. Participation constraint on relationship sets are represented by an integer pair (min:max) on each participating entity set. The value min gives the minimum number of relationship instances in which an entity of the participating entity set is included, while max gives the maximum number of such relationship instances. A max=* indicates no constraint on the maximum number of relationship instances. For example the ADVISES relationship from figure 3.7. A GRAD-STUDENT instance is advised by only one (or none) FACULTY instance, while a FACULTY instance is able to advise many GRAD-STUDENT instances. Another example is the OWNS relationship set, where every VEHICLE must be owned by a PERSON, and to be classified as VEHICLE-OWNER the PERSON must have at least one VEHICLE occurrence.

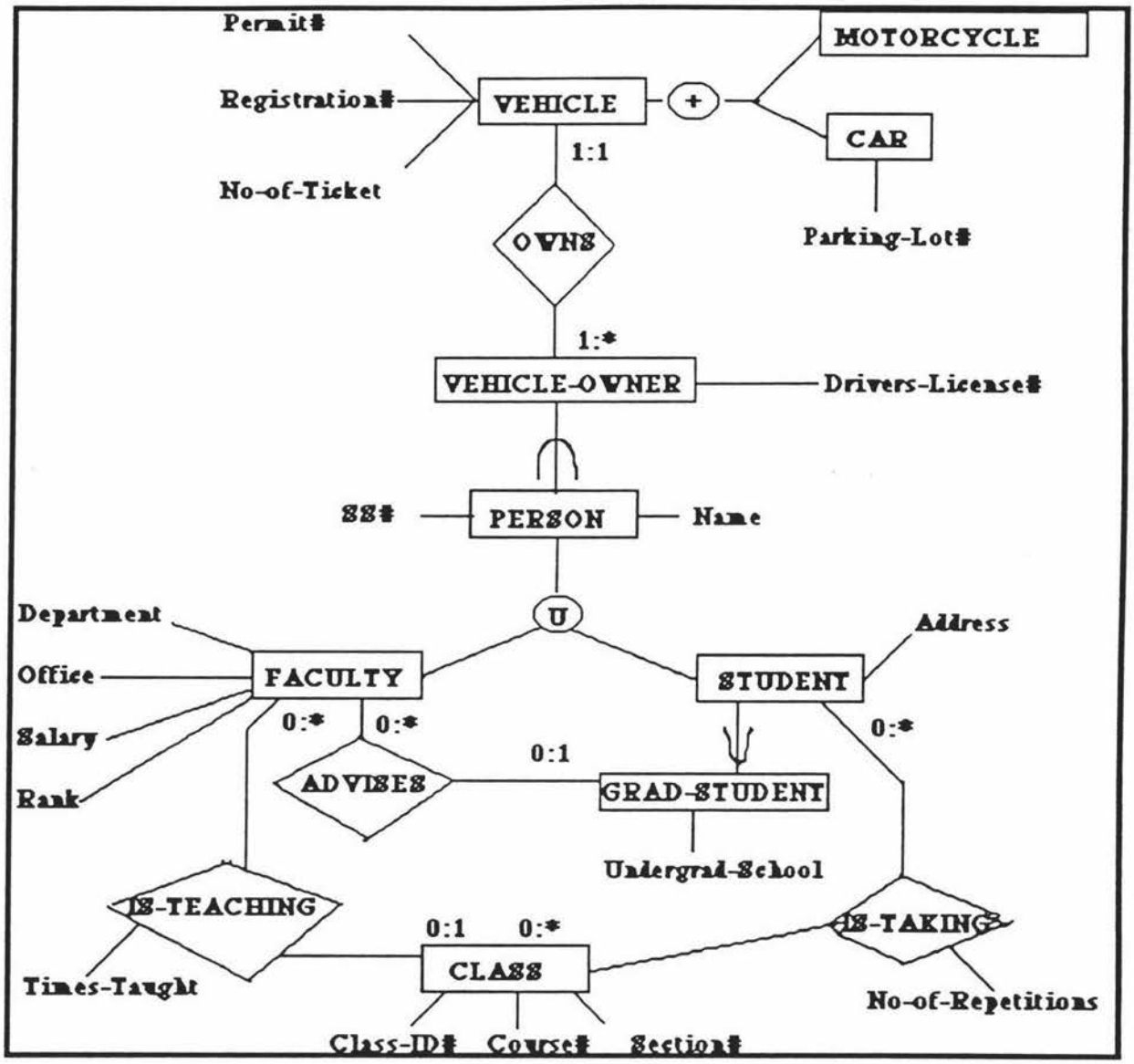


Figure 3.7 A sample ECER schema diagram (Czejdo et al, 1990:28)

Generalisation/Specialisation

"Generalisation/Specialisation in the ECER model defines super-class/subclass relationships between entity sets. Each entity in a subclass entity set must also be an entity in the super-class entity sets." (Czejdo et al, 1990:29)

The authors see the two abstraction mechanisms as the reverse of one another:

"This has also been called an IS-A relationship from subclass (specialisation entity set) to the super-class (generalisation entity set)...GRAD-STUDENT is a specialisation of STUDENT, thus each graduate student is also a student. Conversely, STUDENT is a generalisation of GRAD-STUDENT, thus a subset of the students are graduate students." (Czejdo et al, 1990:29)

The authors define three types of generalisation/specialisation, these and the notations used are:

Type 1: Involves two entity sets, one entity set is a subset of the other, denoted by placing a subset symbol on the arc connecting the two entity sets. GRAD-STUDENT is a specialisation of STUDENT, and VEHICLE-OWNER is a specialisation of PERSON.

Type 2: Involves one generalisation entity set and any number of specialisation entity sets, these latter entity sets are not disjoint. In an ECER diagram this type of abstraction is denoted by a circled union symbol placed onto the joining arcs between the entity sets. The union of FACULTY and STUDENT is PERSON, every person is either a faculty member or a student or both.

Type 3: Similar to type 2, involves one generalisation entity set and any number of specialisation entity sets. These latter entity sets are disjoint and form a partition of the generalisation entity set. It is denoted by a circle-plus symbol, as shown in figure 3.7, MOTORCYCLE and CAR constitute a partition of VEHICLE. Every vehicle is either a motorcycle or a car, but not both.

Attributes

Both entity sets and relationship sets may have descriptor and surrogate-key attributes, these are represented by names and are attached to their respective entity or relationship sets by an arc. Specialisation entity sets inherit attributes from generalisation entity sets. E.g. Name and SS# are attributes (descriptor) of PERSON[7], and are also attributes of FACULTY and STUDENT, and GRAD-STUDENT.

Table 3.4 lists the new terms of the Extended Conceptual Entity-Relationship model.

[7](PERSON,{SS#, Name}, person) is the descriptor for the PERSON entity set, where person is the name of the underlying relation. The relation Person has three attributes, namely, PERSON_ID for the surrogate-key attribute, plus SS# and Name (Czejdo et al. 1990:29).

ECER - SUMMARY OF NEW TERMS
• Specialisation (subset)
• Generalisation (union symbol, not disjoint)
• Generalisation (plus symbol, disjoint)
• Functional relationship (1 : 1 ; 1 : *)
• Non - functional relationship (min:max)
• Descriptor attribute
• Surrogate-key attribute

Table 3.4 New terms of ECER

3.2.7 ECER for Soil Database

"A conceptual design of a soil database for a geographical information system" is the title of Fernandez and Rusinkiewicz's paper, published in 1993.

The authors promise to demonstrate a *"conceptual design of a geographical database of soil information"*. The objective of the study is *"to design a soil database for a geographical information system [sic] (GIS) using the Extended Entity-Relationship model"* (Fernandez and Rusinkiewicz, 1993:525).

The first difficulty encountered was identifying what kind of extended entity-relationship model was used by the authors. This is vital for interpreting the diagramming conventions of the presented schema. The paper under review refers to the Teorey et al.(1986) data model, but concludes that *"We have used the model to incorporate various forms of generalisation and specialisation, including subsets and unions (Czejdo et al., 1990)"* (Fernandez and Rusinkiewicz, 1993:525).

Figure 3.8 shows the conceptual schema of Fernandez and Rusinkiewicz's paper. An examination of the notation highlights the following differences from the ECER (Czejdo et al., 1990) conventions.

The schema presented in the paper under review:

- always shows entity set identifiers (e.g. Polygon-ID)
- groups simple attributes into a composite-attribute, indicated by a double oval (e.g. Physical Characteristic is a composite attribute of the LAYERS entity type)
- uses a double line to indicate that a specialisation is total.

The differences between the notation used in Fernandez and Rusinkiewicz's paper and the notation proposed by Czejdo's et al. are minor, this is why the conceptual schema of figure 3.8 is classified as an ECER schema.

To appreciate the schema in figure 3.8 additional information is needed about the soil map produced by the USDA-SCS:

The soil maps are organised into map units which consist of one or more components. A soil consociation is a group of areas dominated by one type of soil, while a soil complex consists of two or more types of soil. Areas with no soil material are termed miscellaneous. Each individual area on the soil map is a delineation, and has the same dominant component(s) of the map unit to which it belongs. Once a soil map is digitised and the topology is created, map delineations become polygons. In the authors' opinion, these polygons are independent entities and represent soil and terrain characteristics, and belong to one of the three groups:

- (a) consociations, where each polygon is described by one (dominant) type of soil
- (b) complexes, where each polygon is described by two or more type of soils
- (c) miscellaneous, where each polygon is described by information other than soils.

A soil individual is made up of soil layers (horizons) which determine the "type of soil".

After investigation of the notation and the mapping conventions of the schema, figure 3.8 is comprehensible.

The biggest problem of the paper is that it does not present conceptual database design. The paper uses a data modelling technique to describe the structure of a digitised soil-map. The authors model a digitised soil map, not the reality to be represented in a database. Therefore the method is not suitable as a guide in the practical exercise of this thesis.

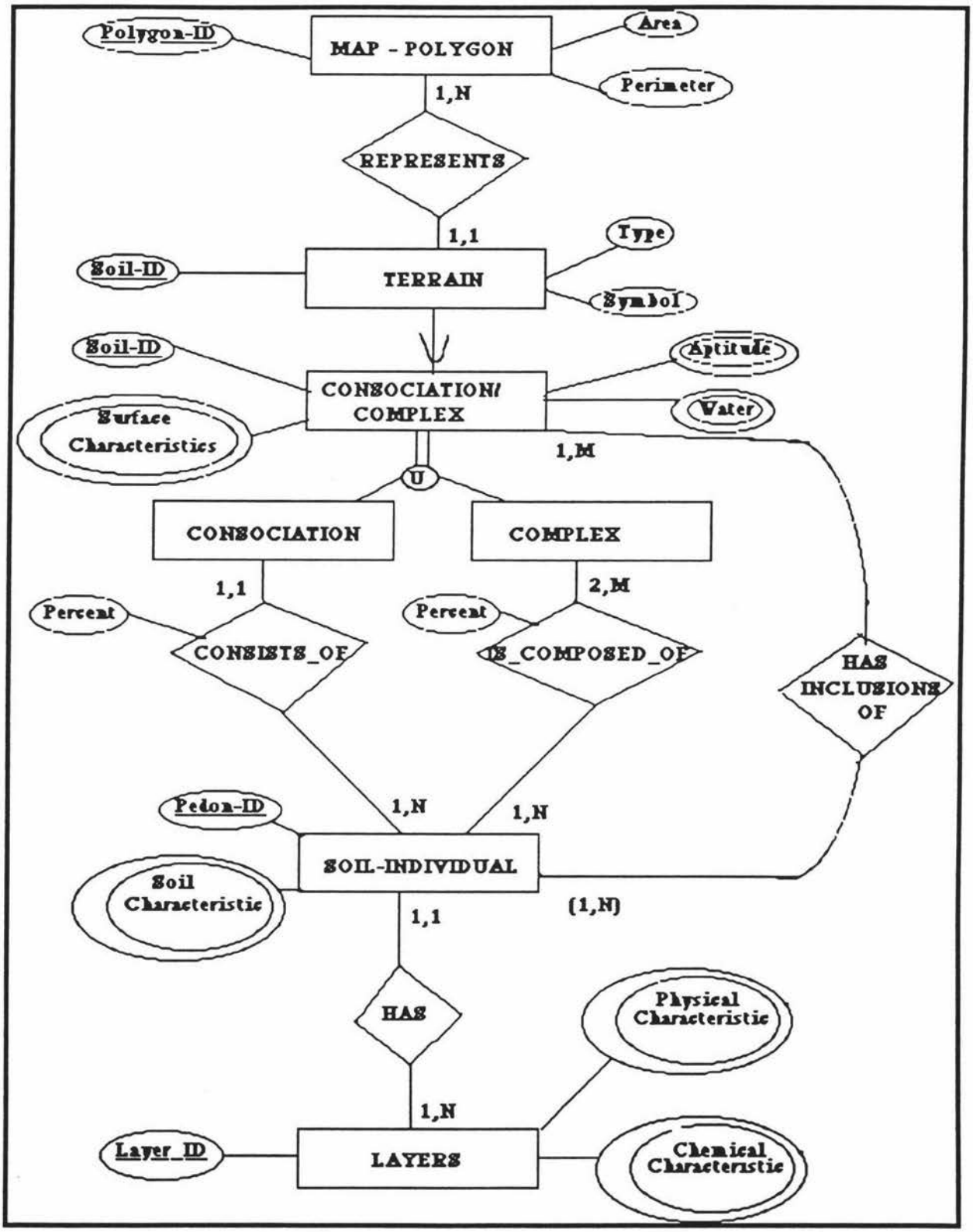


Figure 3.8 Conceptual schema for the soil database (Fernandez and Rusinkiewicz, 1993:530)

3.2.8 Bedard and Paquette's Sub-Model Substitution Method

Bedard and Paquette's paper titled "Extending Entity/Relationship formalism for Spatial Information System, was published in 1989.

In the introduction the authors argue that the ER model is a useful tool, but its expressive power is limited when considering the data needs of spatial phenomena:
"Specialists in non-spatial information systems (e.g. banks, hospitals, schools) have been working with E/R for the last ten years and have recognised the need to improve E/R modelling for special purposes." (Bedard and Paquette, 1989:819)

Bedard and Paquette review the ANSI/SPARC framework, the paper gives definitions of what is termed the French E/R model, identifies some problems related to the modelling of spatial data, and proposes three extensions to the French E/R model to enable modelling of spatial phenomena.

A short inspection of the French E/R model and the problems of spatial data modelling in conjunction with the critique of two presented extensions (i.e. the Sub-Model Substitution technique and the inclusion of Cartographic Only Objects) are included in this thesis.

French E/R model - "Individual Formalism"

"For this paper, we use the French notation called "Individual Formalism."" (Bedard and Paquette, 1989:821) The authors do not give any rationale for their preference of the French E/R model.

"According to the E/R concept, we make conceptual data models by identifying, classifying, describing and relating parts of the real world to organise the information into a formal structure amenable to a computer form." (Bedard and Paquette, 1989:821)

The three basic constructs of entity, relationship and attribute are the same as Chen's original model. The differences are: a relationship set can associate no more than two types of entities, and the cardinality of a relationship set is given by two variables. The two variables show the minimum and maximum number of a relationship in which an entity occurrence takes part.

The graphical representation of these concepts are: an entity type is represented by a rectangle containing its name at the top in uppercase letters. A relationship is represented by a line with a central ellipse containing its name at the top in uppercase. Attributes use lowercase letters and are included either in the rectangle of the entity type or in the ellipse of the relationship. Identifiers are underlined.

The ER Schema in figure 3.9 shows examples: LOT is an entity type which has cadastral# as an identifier and lot area, zoning and land use as descriptor attributes. On the LOT there are many, one or none instances of HOUSE built, giving the relationship set OCCUPY between the two entity types. The relationship set OCCUPY has an attribute called area.

Extensions to the Individual Formalism

The E/R *"formalism is built for a traditional database structure and is not well suited for spatial referencing "* observe Bedard and Paquette.
There are two ways to deal with spatial referencing. The first option is to avoid the *"modelization"* of spatial referencing. The resultant model is insufficient as it does not represent all the available data in the GIS database. The second and right way is, in the authors' opinion, to show all "geometric entities" needed to draw "non-geometric" entities. (The authors do not explicitly define geometric and non-geometric entity types.) The problem with this approach is that it creates an overly complex data model. The authors illustrate this point with an ER schema, shown in figure 3.9. In figure 3.9 "non-geometric" entities such as LOT and HOUSE are related to "geometric entities" like POLYGON.

Including the "geometric entities" raises a problem as *"for most GIS users, geometric entities are not real life objects and have no meaningful attributes"* and it also raises the question of *"which geometric data structure to use"*.

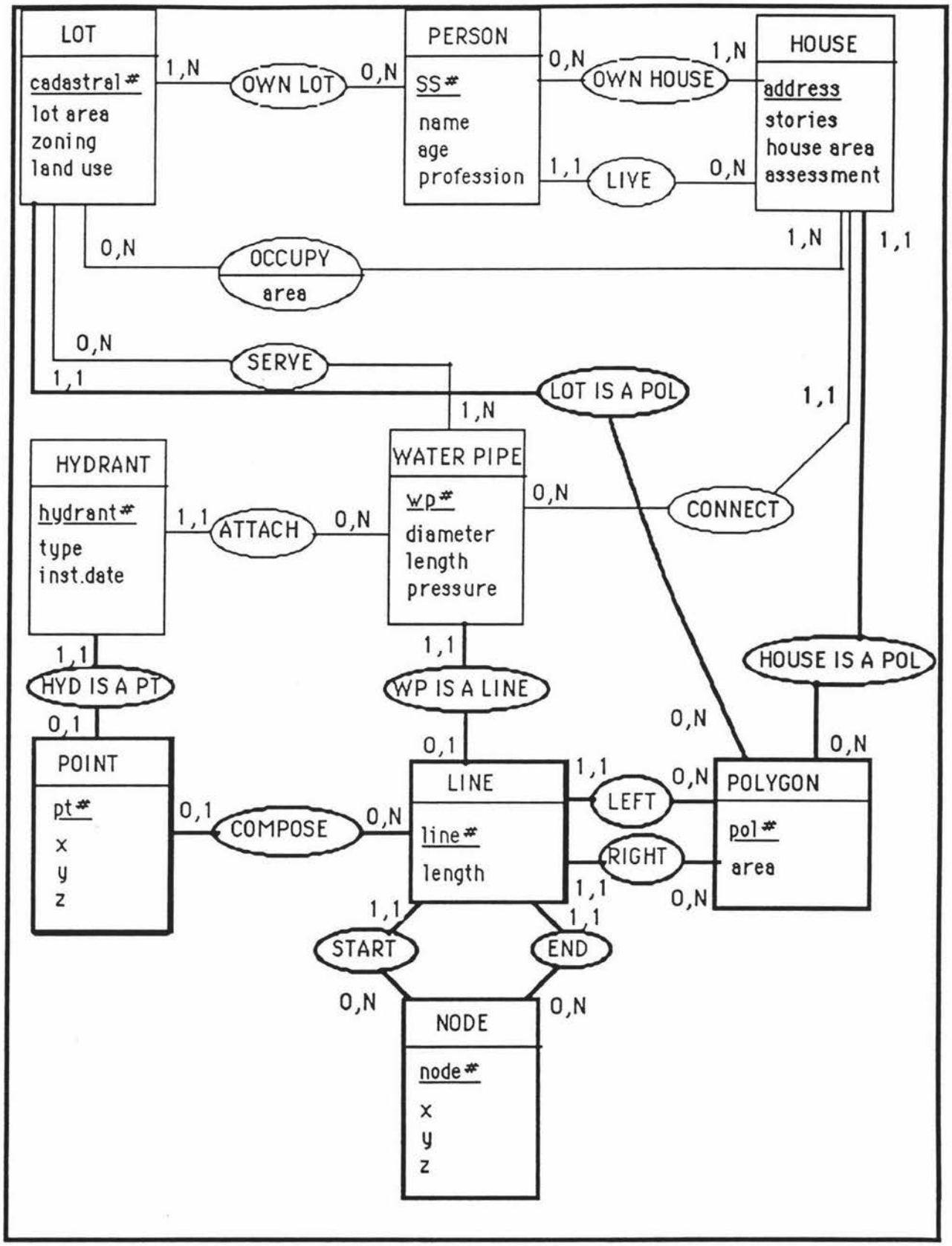


Figure 3.9 ER Schema - explicitly including (in bold) geometric entities and relationships (Bedard and Paquette, 1989:823)

The proposed solution to a manageable schema is to extend the French E/R modelling concepts, by a method called the Sub-Model Substitution technique. Bedard and Paquette suggest that geometric entities and relationship sets to such geometric entity sets,

could be substituted by a symbol placed in the rectangle representing the related non-geometric entity set. (e.g. a line symbol placed in the rectangle representing WATER PIPE, stands for both the WP IS A LINE relationship set and the LINE entity set.)

The Sub-Model Substitution (SMS) symbols with their substituted sub-models are shown in figure 3.10. Figure 3.11 uses the proposed symbols .

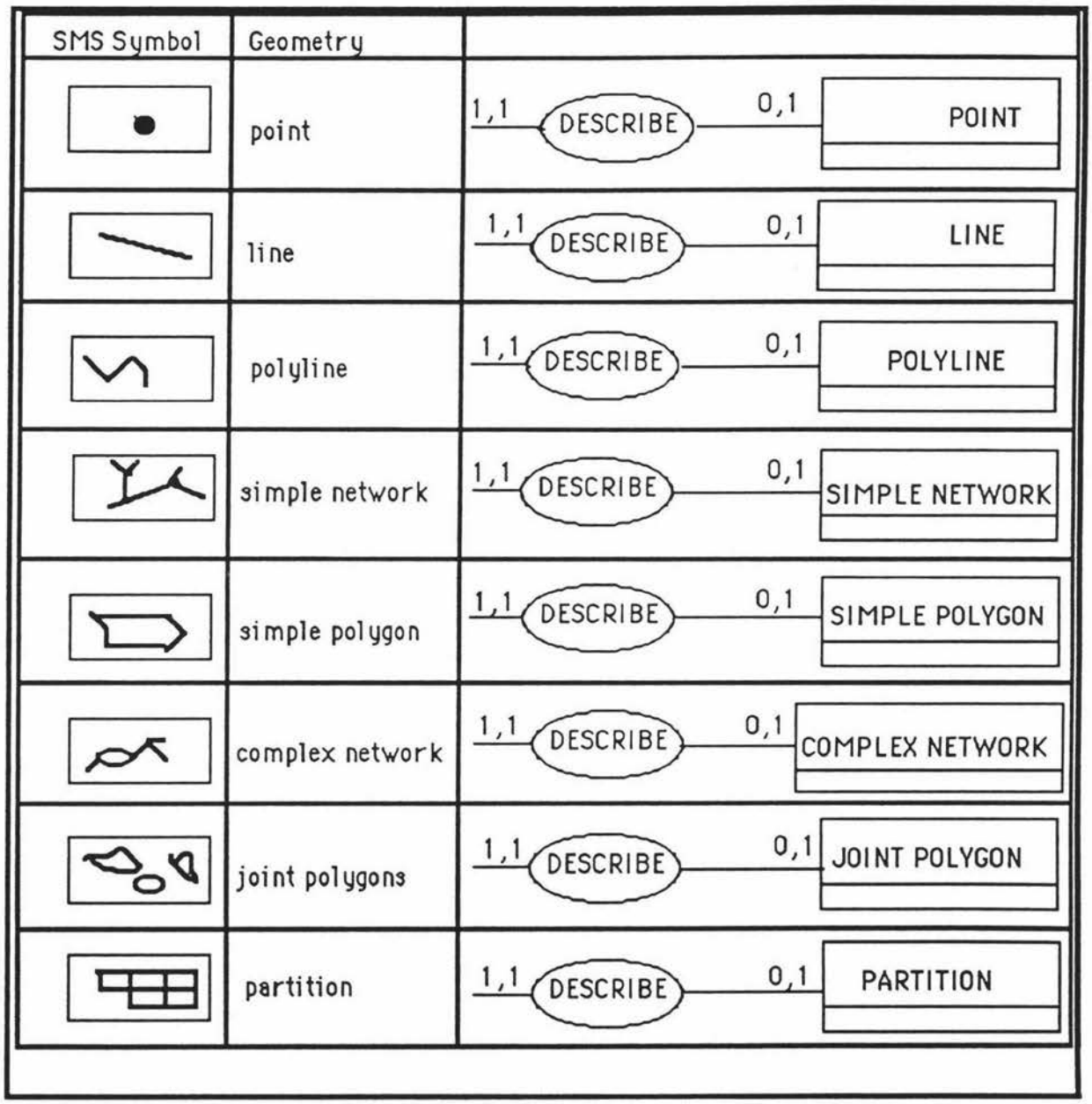


Figure 3.10 The Sub-Model Substitution(SMS) symbols with their substituted sub-models (Bedard and Paquette, 1989:825)

The weaknesses of the SMS technique are outlined by Firns: *"The relationships between "geometric entity sets"* (figure 3.9) *are not represented in SMS schemas...Closer examination of the relationships between geometric entity sets indicates that they are not*

application specific...that part of the model is a meta-model of reality ." (Firns, 1994b:65)

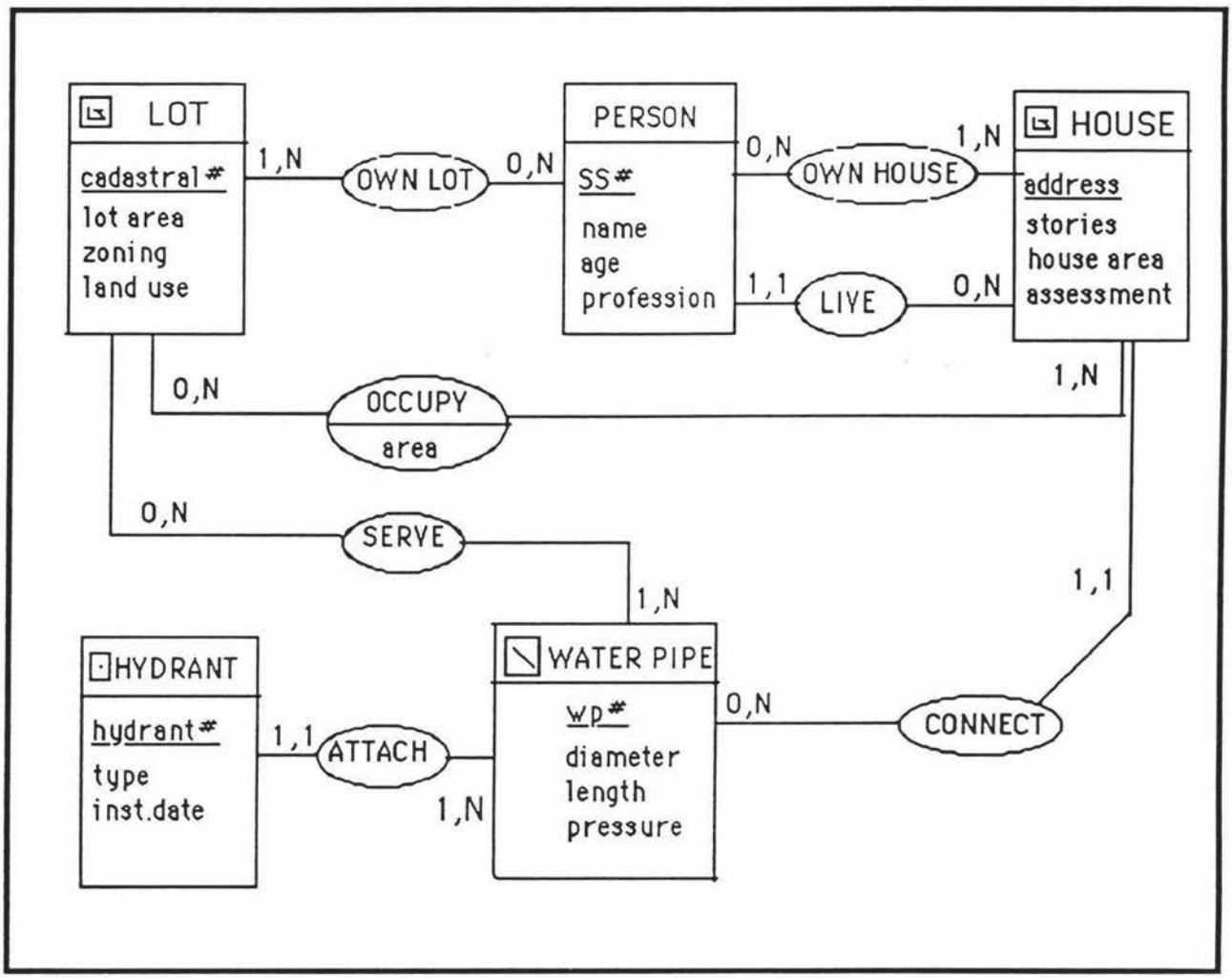


Figure 3.11 SMS Representation of the Model from figure 3.9 (Adopted from Firns, 1994b:64)

"There is an additional problem with traditional E/R modelling: only the entity types explicitly defined in the database are represented" (Bedard and Paquette, 1989:823)

This is a limitation of the model, in the authors' opinion, because SIS objects appearing on output maps may not need database files in the database (e.g. like the legend on topographic maps.) The proposed solution is to differentiate between three types of entities on a French E/R model diagram, i.e. non-mapped, mapped only, mapped and stored entity types. This proposed extension is called the Cartographic Only Object Rule. The potential outcome of using the rule would be a model of the real world objects and their map representation.

Bedard and Paquette's methods model reality together with its geometric representation, therefore the paper does not present an appropriate technique for the case study.

3.2.9 Comparison of ER Models

Numerous extensions to Chen's original ER model exist which enrich the basic constructs (entity-relationship-attribute) in different ways, creating the possibility of confusion among users of these models. Authors take different approaches to remedy the situation. For example: Lenzerini and Santucci (1983:531-537) and Elmasri and Navathe (1989:60-61) give a general summary of new terms, Hainaut proposes a reference model *"according to which any current E-R model can be defined as a specialisation"* (1991:433), Spencer et al. put forward an "ER Standards Proposal" (1991) giving precise definitions of concepts in E-R models.

Table 3.5 compares the original and the discussed versions of the ER model. The basic building blocks of the French E/R model are the closest to the original model's concepts. The French E/R model is not an extended version, but is only a different notation for Chen's model. The ECR and the ECER models extend Chen's model by adding new abstraction mechanisms to it.

These extensions are partly used in the GIS schemas. Armstrong and Densham (1990) utilise the ISA Category concepts and multi-value attributes of the ECR model (refer to section 3.2.5), while Fernandez and Rusinkiewicz's (1993) paper put to use the union type of generalisation concept of the ECER model. The abstraction mechanisms worked correctly, but they did not have an influence on the final outcome of the above mentioned database designs. Not recognising or, recognising but not clearly separating the difference between a real world object and its representation was the major fault of the reviewed papers.

CONCEPT	ER MODEL (refer to 3.2.1)	FRENCH E/R (refer to 3.2.8)	ECR MODEL (refer to 3.2.4)	ECER MODEL (refer to 3.2.6)
Entity	•Entities are classified into different Entity sets.	•Entities are classified into different Entity sets.	•Entity types •Category (ISA) •Category (generalise)	•Specialisation (subset) •Generalisation (union) •Generalisation (disjoint)
Relationship	•A relationship set is an association between entity sets.	•A relationship set is an association between no more than two types of entity set.	•Functional relationship (1:1 ; 1:N) •Non - functional relationship (M:N)	•Functional relationship (1:1 ; 1:*) •Non - functional relationship (min:max)
Attribute	•Entity sets and relationship sets are described by the same set of attributes. •Single-valued	•Entity sets and relationship sets are described by the same set of attributes. •Single-valued	•Basic and Acquired •Single and Multi-valued	•Descriptor •Single-valued
Primary Key	•A unique attribute of the entity set or relationship set.	•A unique attribute of the entity set or relationship set.	•Entity "identifier"	•Surrogate key attribute
Cardinality	•Describes the constraint on the number of entity instances that can participate in a relationship. •Property of relationship.	•Describes the constraint on the number of entity instances that can participate in a relationship. •Property of relationship and is given by two variables.	•Property of entity sets participating in the relationship and is given by two variables.	•Property of entity sets participating in the relationship and is given by two variables.

Table 3.5 Comparison of terms of ER, French E/R model, ECR and ECER Models

3.3 Object-Oriented Approach to Conceptual Data Modelling

The object-oriented paradigm is only well-defined in programming language environments. The interest in the approach has grown rapidly in other information systems areas (e.g. object-oriented development, object-oriented database management systems), but generally accepted definitions are yet to emerge in those fields. An excerpt from Roger King's much cited paper illustrates both the proliferation of interest in the object-oriented approach and the confusion which this hype brings into the database domain:

"It's exciting to see religious fervour grip one's sub discipline. It improves attendance at conferences...however, there is considerable disagreement concerning the definition of "object-oriented"...Obviously, it takes time to iron out common terms...But the problem is that database researchers have recently begun to refine what they meant by object-oriented." (1989:24)

At present, there are standardisation efforts in progress for "general" object-oriented database management systems (Cattell, 1994) and there are publications specifically aimed at spatial object-oriented databases. Survey papers review the object-oriented modelling constructs for Geographical Information Systems (Egenhofer and Frank, 1989; 1992), and summarise object-oriented principles and techniques for geographic data management (Gunter and Lamberts, 1994), but the war of the differing definitions is still on.

Publications on object-oriented data modelling in the spatial context can be found in impressive numbers (e.g. Feuchtwanger, 1987; Armstrong and Densham, 1989; Choi and Luk, 1990; Kemp, 1990; Roberts and Gahegam, 1993). This thesis chooses one from the available research papers (Worboys et al., 1990). The review and critique of this particular paper will illustrate the complexity and potential pitfalls of the object-oriented data modelling paradigm and gives an excellent opportunity to consider many important issues in a succinct manner. Before the discussion of Worboys' et al. paper, background knowledge about the IFO database model will be provided because this model is used in their paper.

3.3.1 Definitions of IFO

The IFO[8] Database Model (Abiteboul and Hull, 1987) has been developed for the investigation of semantic database modelling issues.

The primary components of semantic models to represent the structure of data are:

- the explicit representation of objects, and their attributes and their relationships among objects
- type constructors for building complex types
- ISA relationships
- derived schema components (Hull and King, 1987).

The IFO model incorporates all these principles within a graph-based representational framework .

Objects

The basis of any IFO schema is the representation of object types. There are two kinds of object types: atomic and nonatomic.

Atomic types are: printable atomic types, denoted by a square node (e.g. figure 3.12 (a)), abstract atomic types, denoted by a diamond (e.g. figure 3.12 (b)), and free atomic types, denoted by an empty circle (e.g. figure 3.12 (c).

The printable type corresponds to objects which can be thought of as user input and output, abstract atomic types typically represent a domain, and free atomic types show that a given object type is obtained via an ISA relationship.

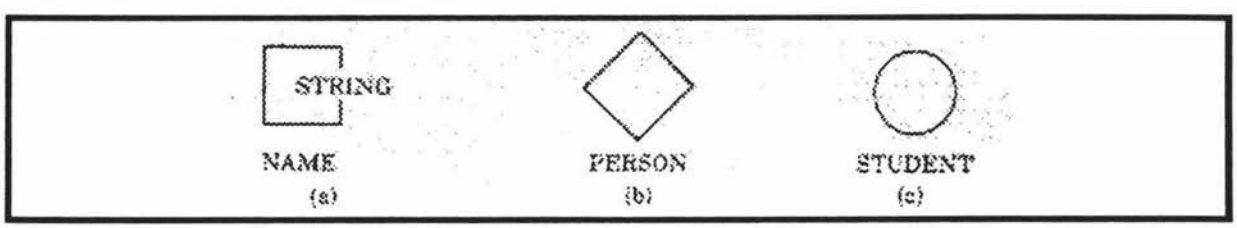


Figure 3.12 Three atomic IFO types (Abiteboul and Hull, 1987:529)

Type constructors

The first of the two abstraction mechanisms for constructing nonatomic object types out of existing types is called "aggregation". This abstraction mechanism is represented by a

[8]Acronym for Is-a relationships, Functional relationships, Complex Objects.

⊠ - vertex ("cross-vertex"). An example is shown in figure 3.13(b), in which the object type MOTOR-BOAT is viewed as being an ordered pair of a HULL and a MOTOR.
The second abstraction mechanism is called "grouping" which forms a new object type from instances of the same object type. For example, figure 3.13(a) shows how a ⊛ - vertex ("star-vertex") is used to depict the type corresponding to sets of students.

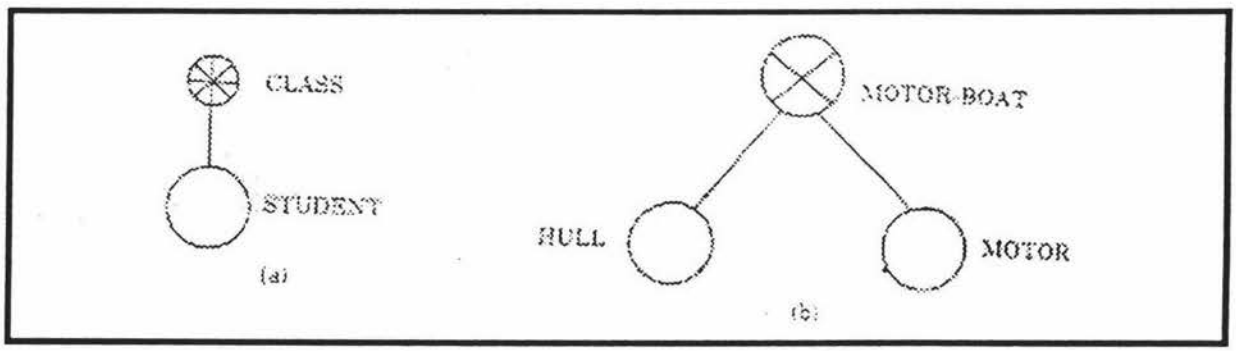


Figure 3.13 Constructed object types (Abiteboul and Hull, 1987:530)

Fragments

In the IFO model functional ("has-attribute") relationships are represented by fragments. A fragment consists of vertices with directed graphs between them. In figure 3.14 there are two simple fragments illustrating how fragments are used to model a set of students, and a set of courses.

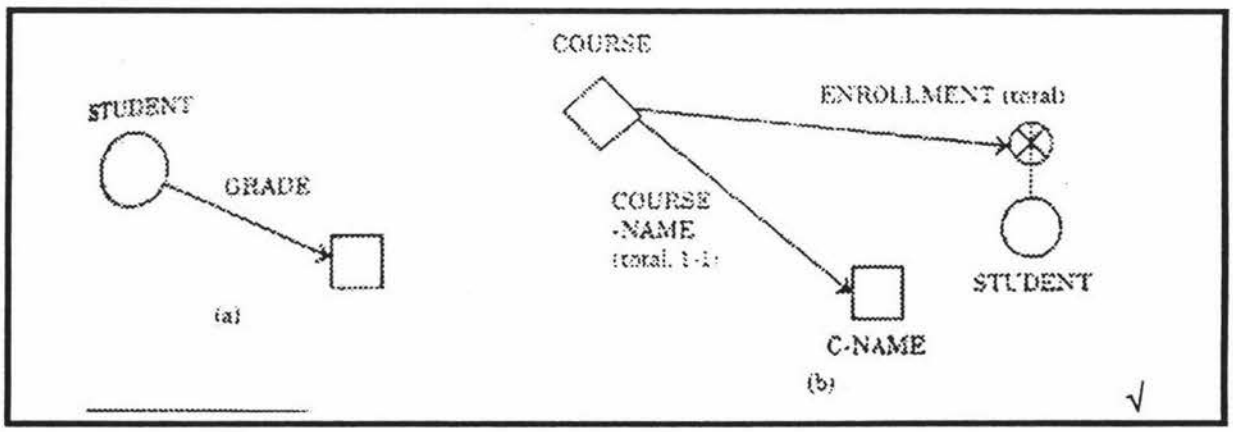


Figure 3.14 Two simple fragments (Abiteboul and Hull, 1987:531)

ISA relationships

"ISA" relationships specify that one set of objects is a subset of another set of objects. An ISA relationship from a type SUB to a type SUPER indicates that each object associated with SUB is associated with type SUPER, and attributes of SUPER are inherited by SUB.

In the IFO model two types of ISA relationship are distinguished: specialisation , depicted by a broad arrow (⟹), and generalisation, depicted by a shaded arrow (➡). Specialisation can be used to define possible roles for members of a given type (e.g. a person might be a student, a mother, a teacher). In contrast, generalisation represents situations where distinct, pre-existing types are combined to form a new virtual type (e.g., the types car and motor-boat might be combined to form the type vehicle).

IFO Schemas

The fragments are combined to form an IFO schema. An example of an IFO schema is provided in figure 3.15.

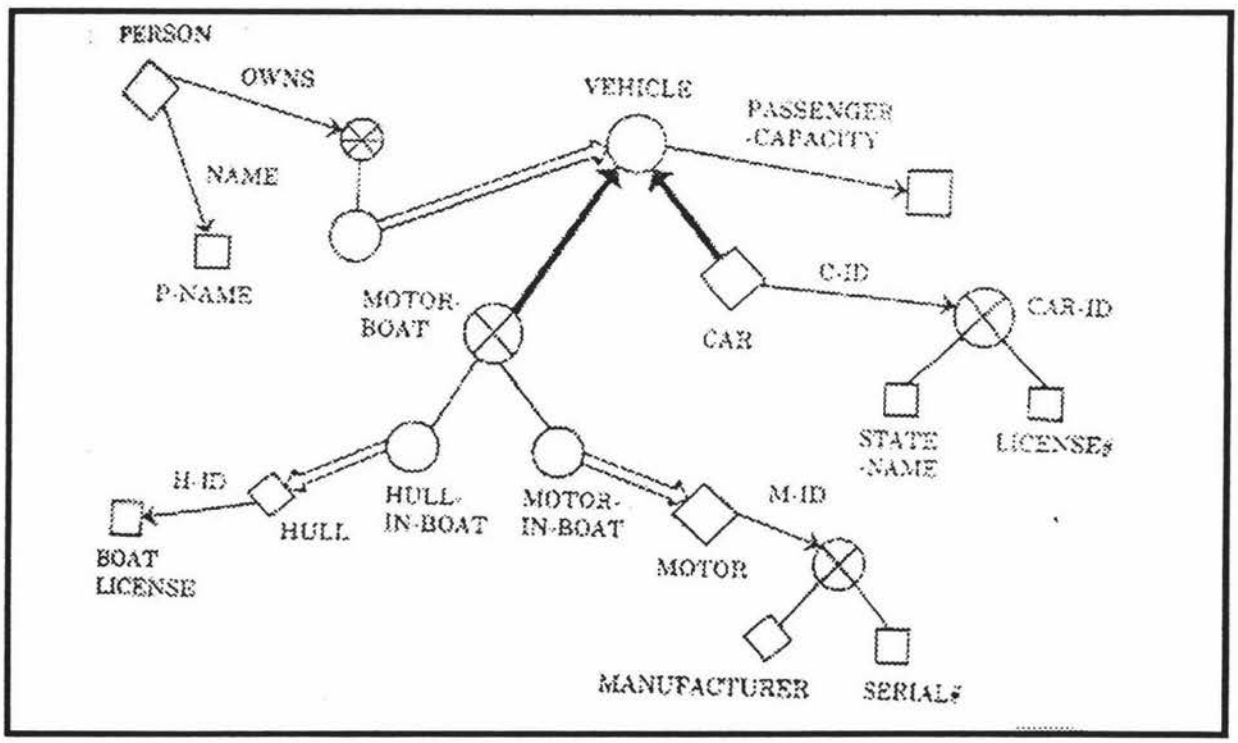


Figure 3.15 The Vehicle example (Abiteboul and Hull, 1987:533)

The Vehicle example, in figure 3.15 can be interpreted as follows: A PERSON (abstract object type)has a NAME and owns a set of VEHICLE. A VEHICLE is a free (precisely generalised) object type of MOTOR-BOAT and CAR object types, passenger-capacity is its only attribute. A CAR (abstract type) has CAR-ID, which is represented as an aggregation of STATE-NAME and LICENCE# which are both printable atomic types. MOTOR-BOAT is a composite object too, it is an aggregation of two object types, both of which are free types and are products of ISA relationships, namely specialisation.

HULL-IN-BOAT is a special type of HULL, MOTOR-IN-BOAT is a special type of MOTOR, HULL and MOTOR are abstract types. HULL has only one atomic type attribute, BOAT- LICENCE, while MOTOR is identified by a composite object type, consisting of the atomic types MANUFACTURER and SERIAL#.

3.3.2 Object-Oriented Data Modelling for Spatial Databases

This section will present and critically evaluate the paper titled "Object-oriented data modelling for spatial databases" by Worboys, Hearnshaw and Maguire (1990).

The authors stated objectives are:

"This article discusses the key concepts in object-oriented modelling and demonstrates the applicability of an object-oriented design methodology to the design of geographical information systems." (Worboys et al., 1990:369)

They also promise to illustrate the object-oriented concepts:

"In order to show more clearly how this methodology may be applied, the paper considers a specific object-oriented data model, IFO. Standard cartographic primitives are represented using IFO." (Worboys et al., 1990:369)

In the introduction the authors stress the importance of choosing the right data model for the problem domain during database design. Different problems (applications) require different means of representation, which is the reason why many database models are described in the literature, argues the paper. Some models are closer to implementation structures, e.g. the relational model, and others are closer to the original problem framed by the user. One of the latter type is the "object-oriented data model" states the paper while promising to elucidate its features. Yet the authors do not define the object-oriented data model and avoid the contentious questions:

"There is no clear definition of an object-oriented data model...However, there is a clear ascending chain from the relational model through the earlier semantic data models to object-oriented models and it is in this context that this paper considers object-oriented modelling methods." (Worboys et al., 1990:370)

The introduction does not specify which phase of the database design is in question, but differentiates between the emerging object-oriented database management systems, as

being the technology, and the object-oriented data modelling, as being the design tool for a database, which is quite unnecessary.

3.3.2.1 Data Models

Following the introduction, the next five sections of the paper describe the concepts of different data modelling techniques and their application in the spatial database design context.

The relational data model

First the relational model is mentioned and it is claimed that for many design problems the relational model lacks expressive power. To prove the point, the paper shows the relational model of a polygon and comments:

"This model of polygon as a set of relations, though complete, is low-level, and some way from one which represents a user's normal view of such an object." (Worboys et al., 1990:371)

POLYGON (Polygon ID, Rind ID, Ring Seq) RING (Ring ID, Chain ID, Chain Seq) CHAIN1 (Chain ID, Point ID, Point Seq) CHAIN2 (Chain ID, Start Node, End Node, Left Pol, Right Pol) NODE (Node ID, Point ID) POINT (Point ID, X Coord, Y Coord)

Table 3.6 Relational model of a polygon (Worboys et al., 1990:371)

In order to understand this model it is necessary to go some pages forward in the paper, where the authors quote the detailed definitions of spatial objects proposed by the National Committee for Digital Cartographic Data Standards in Moellering 1986, summarised by van Roessel, 1987 :

"A point is a zero-dimensional spatial object with coordinates and a unique identifier within the map.

A line is a sequence of ordered points, where the beginning of the line may have a special start node and the end a special end node.

A chain is a line which is a part of one or more polygons and therefore also has a left and right polygon identifier in addition to the start and end node.

A node is a junction or endpoint of one or more lines or chains.

A ring consists of one or more chains.
A polygon consists of one outer and zero or more inner rings."

The relational model in table 3.6 does model a polygon, as defined by the National Committee for Digital Cartographic Data Standards. The real problem of the relational model presented is that *polygon* is an abstraction itself. The relational model of table 3.6 models a model of reality, not the reality.

Entity-relationship modelling

The authors introduce Chen's model, give an example, figure 3.16, and conclude that the ER approach more closely describes the entities in the problem domain as perceived by a human being, compared to the relational model.

Figure 3.16 Entities and Relationship (Worboys et al., 1990:371)

Extensions to the entity-relationship model

For many systems the initial set of modelling techniques provided by the ER model is inadequate. The paper proceeds to introduce these additional abstraction mechanisms by using IFO model notation on GIS examples.
Generalisation is the abstraction mechanism which enables a group of entities of similar type to be considered as a single generalised type. In figure 3.17 SETTLEMENT is a generalised type of TOWN, CITY and VILLAGE.
Specialisation is the abstraction mechanism which defines possible roles for members of a given type. In figure 3.17 entities of type PERSON might be considered occurrences of type ROAD-USER or RAIL-USER.

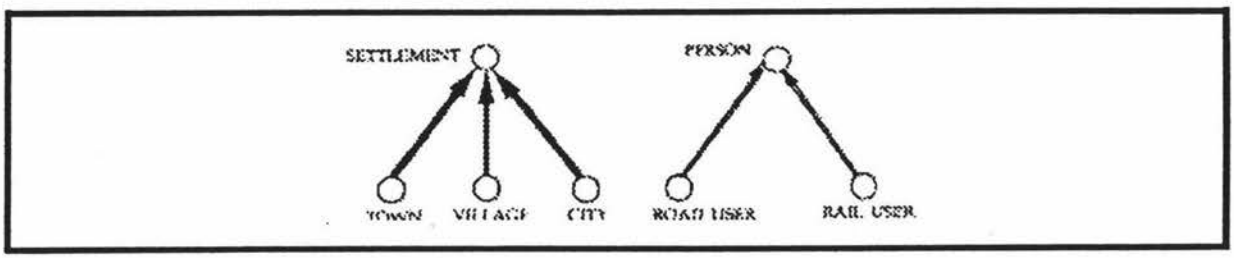


Figure 3.17 Generalisation and Specialisation in IFO (Worboys et al., 1990:372)

In the authors' opinion, the two type constructors are the inverse of each other, the only distinction being that a generalised type inherits its structure from its subtypes, while in the case of specialisation the subtypes inherit their structure from the super-type.

Besides generalisation and specialisation, the authors treat aggregation and association in a way similar to the original IFO model.

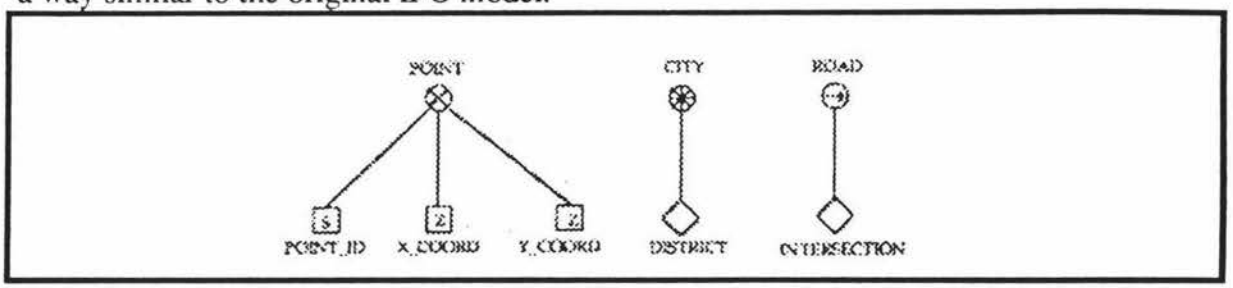


Figure 3.18 Aggregation and Association (Worboys et al., 1990:373)

Figure 3.18 is used in the paper to explain the concepts of aggregation and association (or grouping). POINT is an aggregation of identifier and coordinates. CITY is an association of DISTRICT. ROAD is an ordered association of INTERSECTION.
The ordered association, an ordered collection of objects of the same type, denoted by ⊜, is not originated from the IFO model.

Object-oriented data modelling

For the first time in the paper the authors give an account of their own personal beliefs on the object-oriented data model, in the form of a loose and general description of the concepts.

"In object-oriented data modelling, all conceptual entities are modelled as objects. An abstraction representing a collection of objects with properties in common is called an object type. Objects of the same type share common functions....We have seen how complex types may be formed from primitive types using generalisation, specialisation, aggregation and grouping. These are the primary object type operations in object-oriented data modelling." (Worboys, 1990:374)

The last statement of the excerpt is confusing, further discussion will follow in this thesis. The next quoted paragraph proves that the authors are aware of the real nature of a potential object -oriented database model.

"Object-oriented data models support the description of both the structural and the behavioural properties of a database. Structural properties concern the static organisational nature of a database. Behavioural properties are dynamic and concern the nature of possible allowable changes to the information in the database. This paper concentrates on the structural description." (Worboys et al., 1990:374)

The next sections (6,7) of the paper under review start with the announcement: "IFO *is truly object-oriented in that all its component types may be composite.",* this is followed by a lightweight introduction to the IFO database model.

Figure 3.19 shows the application of IFO to represent the fundamental spatial elements. (An IFO representation of POINT is given in figure 3.18)
NODE is a special type of POINT, with its own node identifier. LINE is modelled as an ordered association of points, with an identifier and 'begin' and 'end' nodes. POLYGON is an ordered association of ring, which in turn is an ordered association of chains. A chain is a special type of line with corresponding left and right polygons.

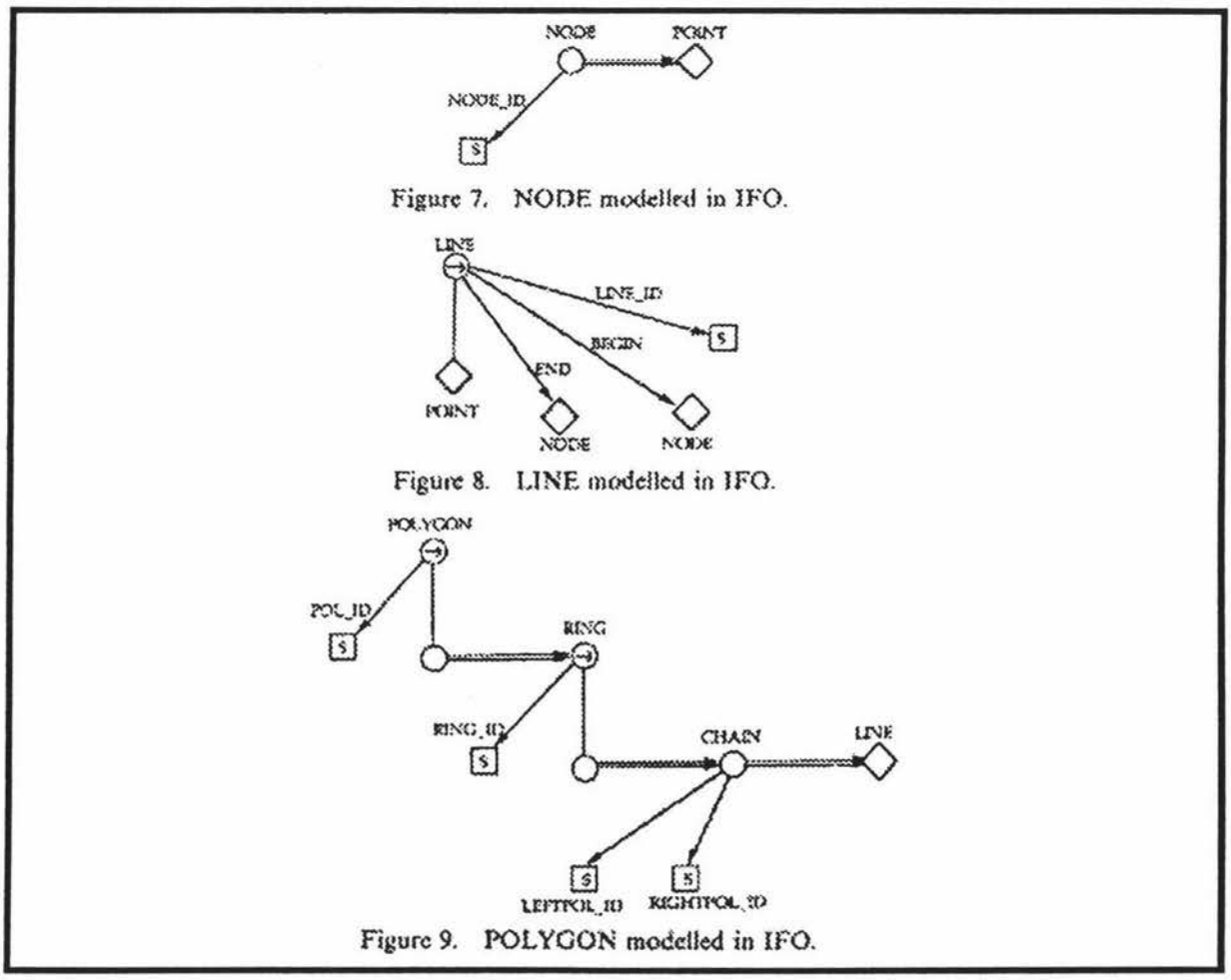


Figure 3.19 Polygon modelled in IFO (Worboys et al., 1990:377)

After modelling the cartographic primitives, the paper applies the IFO notation to represent the structure of a spatial administrative unit on which census data is collected. The data model is shown in figure 3.20, and the authors' explanation to it follows.

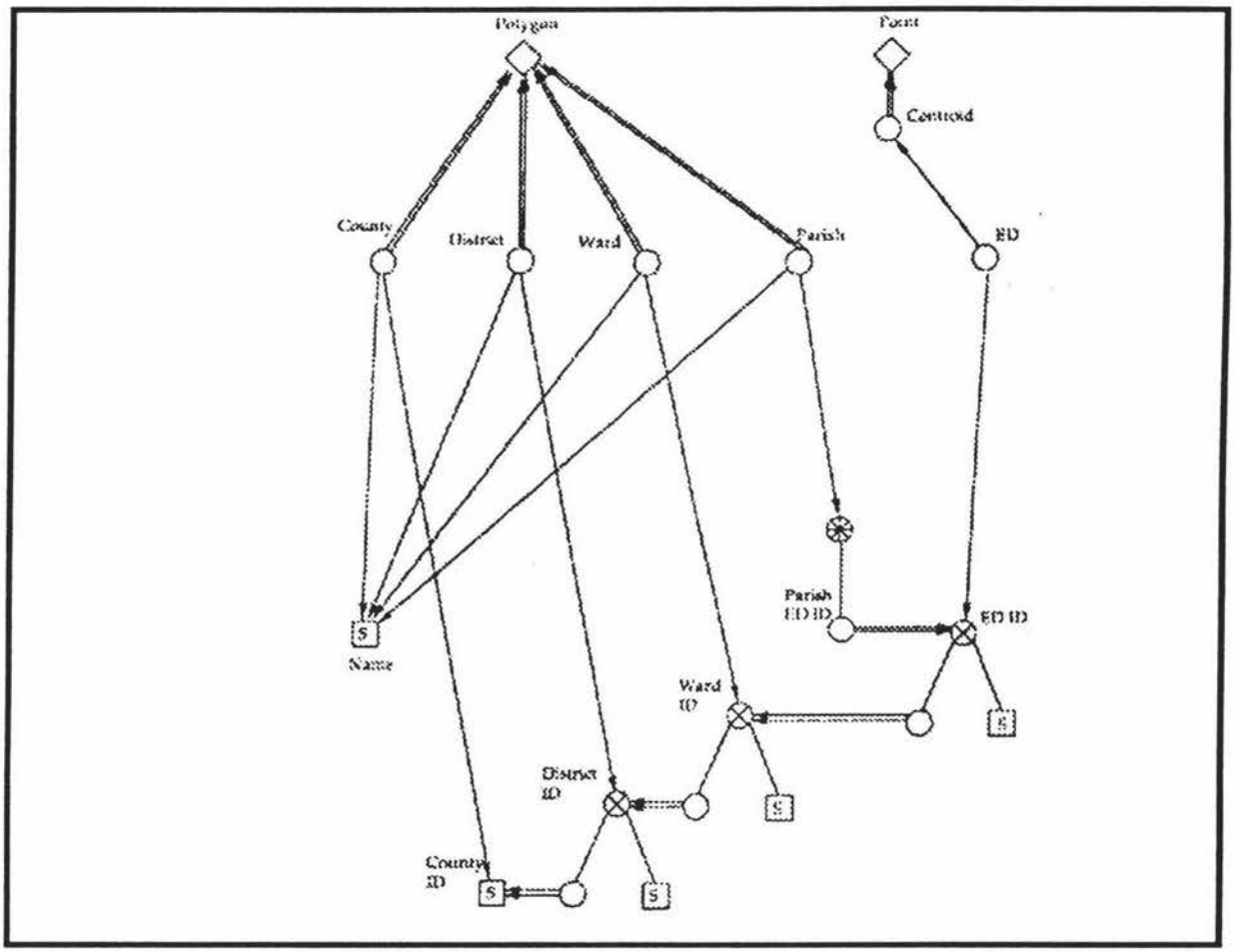


Figure 3.20 Relationship between 1981 census units in IFO (Worboys et al., 1990:379)

The basic unit on which census data is collected is the Enumeration District (ED). The EDs were designed to partition district electoral wards, that is a collection of non overlapping EDs exactly cover each ward. Wards also partition the districts. However, in the case of Leicestershire, those parts which used to be rural districts are also divided into civil parishes.

Each ED is uniquely identifiable by a six character code: two characters for district, two for the ward and two for ED. Each county also has a 2-digit code.

The IFO representation is given in figure 3.20. The relationships between the units are: an ED has a Centroid (for the weighted centre of population, statistical data), which is a specialisation of point. COUNTY, DISTRICT, WARD and PARISH are specialisation of POLYGON. All these units have NAMEs, and identifiers. PARISHes are associated with identifiers for those EDs which are contained in each parish.

Two questions arise from the data model schema shown in figure 3.21. The first is, did the authors apply an "object-oriented" method to produce the schema? and the second is, is this a practical representation of the reality?

3.3.2.2 Discussion of an Object-Oriented Data Model

An object-oriented data model is a set of object-oriented concepts for modelling data.

What is meant by the claim that a concept, or a set of concepts are "object-oriented"? The term object-oriented is used in several disciplines and the explosion of interest in object-oriented approaches has led to many different definitions and interpretations.

A universally accepted definition of an object-oriented data model does not exist, but a set of fundamentally important concepts, the "core model", does. The core model is summarised by Kim (1990:13) as follows.

*"• **Object and Object Identifier.** Any real-world entity is an object, with which is associated a system-wide unique identifier.*

*• **Attributes and Methods.** An object has one or more attributes and one or more methods which operate on the values of attributes.*
The values of an attribute of an object are also an object. An attribute of an object may take on a single value or a set of values.

*• **Encapsulation and Message Passing.** Messages are sent to an object to access the values of the attributes and methods encapsulated in the object. There is no way to access an object except through the public interface specified for it.*

*• **Class.** All objects which share the same set of attributes and methods may be grouped into a class.*
An object belongs to only one class as an instance of that class.
A class is also an object; in particular, a class is an instance of a metaclass.

*• **Class Hierarchy and Inheritance.** The classes in a system form a hierarchy or a rooted directed acyclic graph, called a class hierarchy, such that, for a class C and a set of lower-level classes {Si} connected to C, a class in the set {Si} is a specialisation of the class C, and conversely the class C is a generalisation of the classes in the set {Si}. The*

classes in {Si} are subclasses of the class C; and the class C is a superclass of the classes in {Si}.
Any class in {Si} inherits all the attributes and methods of the class C and may have additional attributes and methods.
All attributes and methods defined for a class C are inherited into all its subclasses recursively.
An instance of a class S is also a logical instance of all superclasses of S."

In the case of an object-oriented database this core data model needs extension. The main difference between programming languages and databases of the same paradigm is that the latter requires the existence of persistent objects stored permanently in secondary storage. The core model does not capture concepts that are important to many types of database applications. Two of the most important such modelling concepts are:

• **Composite Objects**. A composite object is a heterogeneous set of objects, the is-part-of relationship is an aggregation relationship between an object and other objects it references.

• **Versions**. A versioned object is a set of objects which are versions of the same conceptual model; a versioned object consists of a hierarchy of objects which captures the version-of relationship between an object and another object derived from the object. Version control is one of the most important data-modelling requirements in newly emerging database applications.

Many authors agree with Kim's interpretation of object-oriented concepts, Elmasri and Navathe (1989), Date (1991), Rabitti et al. (1991), Sciore (1991), Tagg (1992), Jackson (1991). As early as November 29, 1988. during a seminar presented by Max Egenhofer (Frank and Egenhofer, 1988) on object-oriented database technology for GIS, an object was defined, as *"an encapsulation of data and operations applicable to it"*.

It must be noted, that recently there are significant standardisation efforts in progress by various standardisation groups (i.e. application-specific standards groups and formal standards bodies, Cattell ed., 1994:165), with published results. These publications intend to document the progress, and encourage wider feedback from the database community. One of these publications is the "Object Database Standard: ODMG - 93, Release 1.1"(1994) book, edited by Cattell. Each co-author is a technical representative of the Object Database Management Group (ODMG), a consortium of companies that

trade in object-oriented technology set-up to establish an industry-wide agreement for object-oriented database technology.

"Please keep in mind that this document represents work in progress." - writes Cattell in the preface of the book (Cattell ed., 1994). Chapter 2 includes the data model standard. It is interesting to read the "standardised definition", and to establish the identity of the two proposals:
"The Object Model is simply summarised:

- *The basic modelling primitive is the object.*
- *Object s can be categorised into types. All object s of a given type exhibit common behaviour and a common range of states.*
- *The behaviour of objects is defined by a set of operations that can be executed on an object of the type, e.g., you can "format" an object of type Document.*
- *The state of objects is defined by the values they carry for a set of properties. These properties may be either attributes of the object itself or relationships between the object and one or more other objects." (Cattell , ed., 1994:13)*

Inheritance is discussed under Types and Instances subtitled as: *"A subtype inherits all of the characteristics of its super-types"* (Cattell, 1994:14)

Figure 3.21 is taken from Cattell (1991) it shows an "object-oriented" data model schema, to facilitate the comparison with Worboys' et al. data model schema in figure 3.21.

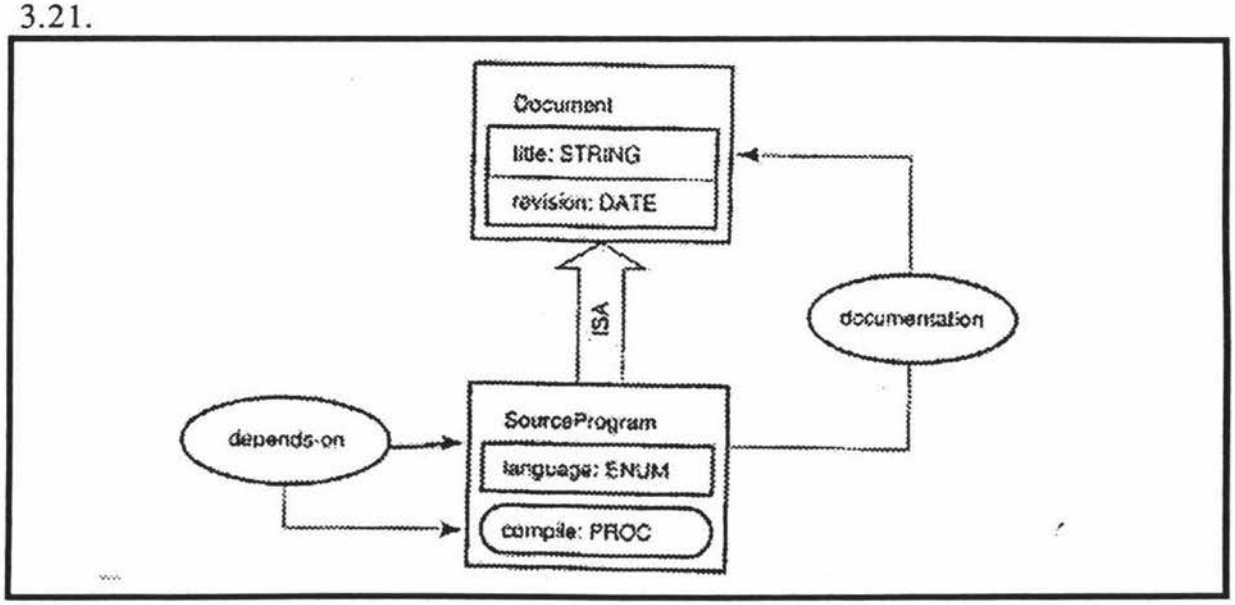


Figure 3.21 An object-oriented data model schema example (Cattell, 1991: 13)

In figure 3.21 an object type is shown as a box, inside a box there are a rectangle for attributes and a rounded rectangle for operations. The ovals and associated arrows represent the type of relationship between the object types. Subtypes are indicated by the wide ISA arrow. Figure 3.21 can be interpreted as : A source program is a type of document; it may depend on other source programs and reference a document that explains what it does. The source-program object also has a procedure (method) field, which contains the commands to compile and link for program execution.

If the two data model schemas in figure 3.20 and 3.21 are compared, the absence of "methods" from the previous schema is apparent. But the question, is IFO "object-oriented"? must be studied in more depth.

3.3.2.3 Is IFO Object-Oriented?

The IFO database model is not object-oriented and has never claimed to be, it is a semantic database model. If the terminology associated with the data model is examined, there is no contradiction to be found until Worboys' et al. paper. The original proposal on the IFO database model, first published by Abiteboul and Hull in 1984 and updated in 1987, classifies the model as a formal semantic database model. Between those publications, one of the developers, Richard Hull classified the IFO model as an object-based semantic database model (Bryce and Hull, 1986).

However, the important point is not how database models are labelled but their components, the tools they provide for the database designer. A comparison between the basic elements of the IFO database model and an object-oriented database model (as it is seen above) highlights some basic differences:

An IFO entity is not equal to an object, an **object** is an **encapsulation of data and methods**. An entity only encapsulates the structural aspects of a real-world concept, it does not imply exclusion of the behavioural aspect of data to be modelled. Peter Chen (in Penul and Tjoa, 1992:1) writes: *"...in the ER world view, data and process are on a equal footing in the past the process part was not emphasised since it was believed that if the data part is done right, the process part can be done easily."*

Inheritance - In object-oriented data modelling both data and methods are inherited, but the IFO model contains only structural inheritance.

Composite Objects - there are not as many type constructors in an object-oriented data model as in IFO; association (grouping) does not exist, the word is used in that context for the concept of a "relationship" between objects. *"In terms of modelling power, we view a core object-oriented data model largely as a subset of a semantic data model; of course, semantic data models lack methods."* (Kim, 1990:35)
King (1989:28) arrives at a similar conclusion: *"Object-oriented models focus on the definition and inheritance of behavioural capabilities, in the form of operations embedded within types, and also support simpler capabilities for structuring complex objects."*

IFO is not an object-oriented database model and it is difficult to defend the authors' point of view. It can be seen from the referenced literature, that when the authors wrote this paper there were quite a few major survey papers around "to clear the air" in the confusion of definitions. The authors randomly refer to elements of an object-oriented database model (Worboys et al., 1990:374 and 381), as described in this thesis. Unfortunately the paper is somewhat loose on definitions and does not fulfil their stated objectives on page 382: *"This paper has traced the development of data modelling from the relational model to a contemporary object-oriented method...object-oriented modelling allows database designers to incorporate more readily the complexities of spatial data."*

3.3.2.4 Evaluation of Worboys' et al. paper

The research described in this thesis aims to find a suitable conceptual data model for the design of GIS databases. The thesis applies the data modelling objectives (Firns, 1990) as the criteria to be met in the process. The first criterion is, as it is interpreted in this thesis, that the given data modelling technique must model reality separated from its representation in a spatial context. The conceptual schema of figure 3.20 presented by Worboys et al. models polygons and points together with real world objects, hence the data modelling technique utilised in the paper is not appropriate for the case study.

3.4 Conclusions

None of the papers on GIS database design address the problem of modelling real world phenomena versus its representation at conceptual level. The inadequacy of various papers was not so much in the level of support from abstraction mechanisms provided by the applied data model, rather the authors' confusion between the genuine real world and its geometric construct which is part of the representation of the real world.

3.4.1 Comparison of Data Modelling Paradigms

The fact that each data modelling paradigm could be used to support the data abstraction needs of GIS is illustrated by the following table 3.7. As it is stated in chapter 1.2.3, the primitive abstraction mechanisms are: classification, aggregation, generalisation, specialisation and grouping (Batini et al., 1992). Table 3.7 uses these abstraction mechanisms to illustrate the point.

Abstraction	Support for Abstraction in ER Model	Support for Abstraction in the IFO Model, a Semantic Data Model	Support for Abstraction in the Object-Oriented Model (Cattell, 1994)
Classification	Real world phenomena are classified into entity sets and relationship sets.	Real world phenomena are classified into abstract object types.	Real world phenomena are classified into object types, encapsulating data and methods.
Generalisation	This is not supported by the original ER model but subsequent extensions have supported generalisation, though definitions differ in various extended models*.	The model supports the abstraction mechanism.	The model supports the abstraction mechanism.
Specialisation	This is not supported by the original ER model but subsequent extensions have supported specialisation, though definitions differ in various extended models*.	The model supports the abstraction mechanism.	The model supports the abstraction mechanism.
Grouping	Grouping is not explicitly supported by the ER model, though support for many to many relationships alleviates this potential shortcoming.	The model supports the abstraction mechanism.	Grouping is not explicitly supported by most Object-Oriented models, though it can be supported using relationships.

Table 3.7 Comparison of Data Modelling Paradigms (Adapted from Firns, 1994b:37)

*This thesis adopts the following definitions for these terms:

Generalisation: *"An abstraction whereby the union of n (>1) disjoint sub-classes forms a super-class*[9]*. That is, the sub-classes partition the super-class."* (Firns, 1994b:27)

[9]The letter 'n' stands for existing types.

Specialisation: *"An abstraction whereby n (>0) possible roles for members of a given entity set are defined as sub-classes (Abiteboul & Hull, 1987). Specialisation sub-classes will not necessary be disjoint and their union will be less than or equal to the super-class."* (Firns, 1994b:27).

The comparison of data modelling paradigms presented in table 3.7 aims to show that the basic modelling power of these data modelling paradigms is very similar. The only differences between the paradigms is the varying definitions applied to the abstraction mechanisms.

3.4.2 Data Modelling Shortcomings Encountered in the Literature

The qualifiers "spatial" and "non-spatial" applied to entities to differentiate them during spatial database design (e.g. Laurini, 1991). Firns (1994b:87) points out that "*in entity relationship modelling the term 'entity' has no semantics attached to it other than that it is something about which data is to be stored*", therefore, without explicit alteration to the original model, the distinction between spatial and non-spatial entities is invalid. At present there is no explicit alteration to the ER model in the literature which would permit usage of the term "spatial entity" when modelling spatially referenced data.

Using the notion of the original relationship set to model spatial relationships (e.g. Wang and Newkirk, 1988), as Firns (1994a:289) observes, *"leads to inconsistencies and/or ambiguities, the result being a lack of rigour in the conceptual design"*. There is a need for the explicit extension of the original model to be able to incorporate spatial semantics in a meaningful and practically useful way (i.e. conforming to the objectives of conceptual data modelling).

"Are points, lines, chains and polygons entities and can they be used to model topological relationships between real world phenomena?" Firns (1994b:44) concludes, that points, lines, etc. are abstractions, the geometric representation of some real world phenomena. Models *"representing a topological structure between objects such as points and polygons, which in turn represent other objects, is a meta-model of reality."* Firns observes, that this type of modelling serves little purpose in the context of conceptual database design, for advocators of this trend in the literature do not model reality. The result of this type of exercise *"provides no useful input to database design - a predetermined spatial data structure is being used to store spatial data"* (Firns, 1994b:45).

"How about models incorporating both real world phenomena and a topological model of these real world phenomena ?" - These models are a combination of two different levels of abstraction (Firns, 1994b:44) and confuse any attempt to model reality.

3.4.3 Summing Up

This chapter presented data modelling design examples from the literature and definitions of the various abstraction mechanisms used in data modelling endeavours were enumerated.

First the chapter focused on the original ER model, then three enhanced versions of this model, the ECR, the ECER, and the French E/R model are presented. The basic elements i.e. entity set, relationship set, single-valued attributes, keys, and the extensions to the basic concepts, i.e. multi-valued and composite attributes, generalisation and specialisation were covered. The various extensions to the original ER data model were compared. Finally, the object-oriented paradigm and a semantic data model, the IFO model were examined. The discussion enabled a comparison between the various data modelling paradigms to be made.

Attempted adaptations of these general purpose data models were considered. The literature review concluded that the originally proposed conceptual data models may be adequate tools, but none of the presented "GIS" conceptual schemas complied with the conceptual data modelling objectives. The schemas generally represented not only real world objects, but some geometric model of reality as well. Hence these data modelling techniques will not be applied in this thesis.

Chapter 4

A Spatially Extended Entity Relationship Model

4.1 Introduction

The research described in this thesis sets out to find a conceptual data model which has the potential to meet the objectives of GIS database design. The data model considered applicable for the design of the case study is introduced in this chapter. The model is the spatially extended entity relationship (SEER) model, an extension to the generic entity relationship technique for the design of spatially referenced databases. The SEER model has recently been proposed by Peter G. Firns (1994a, 1994b) and it can be classified as a conceptual data model in a taxonomy of spatial data models (refer to chapter 2.4).

In this chapter the need for a spatial conceptual model and the requirements for abstraction mechanisms in the spatial context are discussed, followed by an introduction and critique of the components of the SEER model. Finally an evaluation of the SEER model with respect to data modelling objectives is included.

4.2 Rationale for the Extension to the Generic ER Technique

Firns observes that conceptual data models developed for non-spatial database environments are not consistently adaptable to the design of every aspect of SIS databases. The traditional data modelling mechanisms have limitations when modelling spatial concepts. There is currently no published spatial data model analogous to the entity-relationship model, argues Firns. Therefore there is a need for a model which explicitly represents the spatial semantics within conceptual schemas. Firns addresses this problem in his paper (1994a) and PhD Dissertation (1994b). Minor differences exist between the two publications, and the latter work is used as a reference throughout this thesis.

Firns delimits his area of research, that of spatially referenced databases. The definition[1] of a spatially referenced database is given as follows:

---

[1]The definition is not in conflict with the earlier given definitions of SIS and GIS databases.

"A spatially referenced database is defined as a database with the characteristics :

- *It incorporates spatial data* (as defined in 2.3.1 in this thesis).
- *Some spatial relationship types may be represented in the database structure.*
- *It incorporates the type of data typically stored and managed within traditional (non-spatial) databases.*
- *Some non-spatial relationships may be incorporated in the database structure."* (Firns, 1994b:12)

Spatially referenced databases consist of two conceptionally and physically distinct components (i.e. objects with both spatial and descriptive attributes must be represented in two distinct forms, refer to chapter 2.3.3). This characteristic of a spatially referenced database creates an additional level of complexity for database designers. It seems that one object is to be modelled twice for the same database. Firns states that this additional level of complexity could be managed if there existed a conceptual model which integrated the modelling of the spatial and descriptive data aspects of an object.

Another new concept is the notion of spatial representation. Firns states that the inclusion of the general concepts of raster and vector models at the conceptual level in a spatial context would assist applicability of the traditional database design phases (refer to chapter 2.4).

Before the introduction of further new concepts a few terms must be clarified.

It is very important to note that in Firns' work the terms **entity/object** or **entity set/object type** are used interchangeably, and defined as *"some real world phenomena about which data will be stored"* or groups of like objects respectively (Firns, 1994b:86). For example, **polygon** is neither an entity or entity set, but an abstraction. The term **spatial entity** or its synonyms are not used in this context (e.g. Wang and Newkirk, 1988; refer to chapter 3.2.2).

4.3 Requirements for Conceptual Data Modelling Abstractions

Firns undertakes a study to define the requirements of conceptual data modelling abstractions applicable to the design of spatially referenced databases. The requirements fall into two categories. The first category contains the non-spatial abstraction mechanisms that support modelling of those aspects of reality usually supported by conceptual models. The second category contains the abstraction

mechanisms which are needed to model spatial concepts, an original feature of Firns' work. The essential spatial concepts are classified into two groups, the first of which is concerned with how objects relate to space and the second group with how objects relate to each other in space.

4.3.1 Locational and Geometry Requirements

The ability to model objects as they relate to space is a fundamental issue in spatially referenced databases. On the conceptual level, Firns argues that two spatial characteristics of the entities have to be modelled: their location in space and their geometric data types.

4.3.1.1 Location

"Every object or thing which exists at any particular time, does so at some location, that is somewhere in space. Similarly, every event which occurs, does so at some location." (Firns, 1994b:89)

In Firns' opinion there is no possible way to specify the absolute location of phenomena (objects, things, events), some form of spatial referencing technique is required (e.g. coordinate system). Even in reality, the location of a phenomenon cannot be described without reference either to coordinate systems or to the location of other objects.

Firns' approach is to represent the location of phenomena as an entity set, to which other entity sets may be related. Firns here uses a different definition of an entity set from his generally used *"some real world phenomena about which data will be stored"* definition. In a case of representing location as an entity set, Firns applies the term entity to any phenomenon which may be described by data.

4.3.1.2 Geometric Data Types

The second spatial characteristic of an entity set to be modelled is its geometric data type. This is a special term in the spatial context, proposed by Firns. A geometric data type is an example of the class attribute as it is understood in conceptual data modelling. The geometric data type denotes the way in which instances of locationally referenced entity sets will be geometrically represented in a spatially referenced database (e.g. point, line, region). For example: if the geometric data type or class

attribute of the Lake entity set is determined as polygon, this would be associated with every instance of the entity set.

Although the two spatial characteristics are dependent on an assumed model of some kind (i.e. some coordinate system and the existence of a taxonomy of geometric data types respectively) modelling of them, in Firns' opinion, will provide the basis for an independent representation of the objects from the data structure used during the implementation.

4.3.1.3 Thematic Layer as a Conceptual Model

Firns has developed a new conceptual data modelling concept to represent location at the conceptual level - the thematic layer (the "map" qualifier is dropped). The fact that any number of specific classes of spatially distributed phenomena can be considered and represented in isolation from all other classes in distinct map layers has led Firns to recognise this new framework. The term is central to an understanding of the semantics and diagramming notation of the new spatially extended conceptual data model.

The term 'thematic layer' is defined as:

"...a conceptual model by which it is possible to represent a spatial extent as comprising multiple, independent, spatially distributed themes. An instance of a thematic layer corresponds to a single theme, but may comprise different types of phenomena which are, in some way, directly related to each other." (Firns, 1994b:96)

The qualification **directly** means that it is insufficient to include in the same thematic layer phenomena that are only related by spatial coincidence. There must be spatial and/or non-spatial relationships amongst the different types of phenomena being modelled in such a thematic layer.

4.3.1.4 Inter-Layer Connections

Inter-layer connections explicitly represent the existence of multiple thematic layers covering the same spatial extent. Firns draws and extends an analogy between the concept of multiple thematic map layers and "parallel linkages", a commonly occurring structure in generic ER models (Kennedy, 1993:73-76). Figure 4.1 illustrates the

different ways in which a pair of 1:n relationships can connect three given entity-sets in a generic ER model, the "parallel linkages" are shown in figure 4.1(iii).

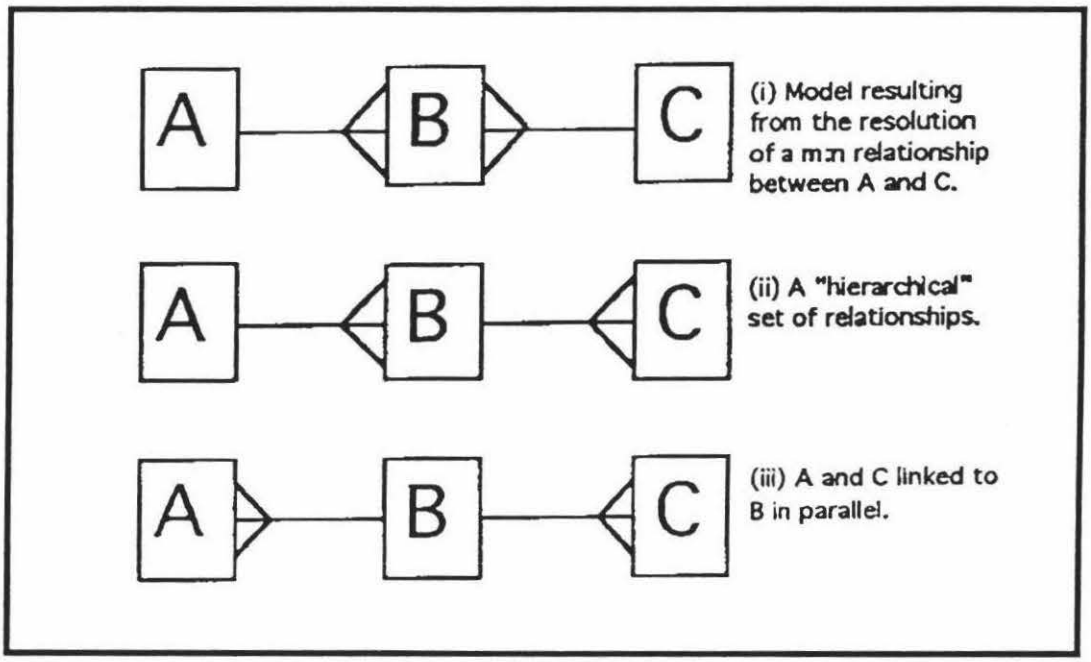


Figure 4.1 Generic Models of Two 1:n Relationships and Three Entity-sets (Firns, 1993:11)

"There is an analogy between entity-sets connected by parallel linkages and thematic layers related to the same spatial extent - as B entity-set in figure 4.1 (iii) is analogous to the spatial extent referred to in the definition of the thematic layer. This is because each thematic layer covering that spatial extent is independent of all other thematic layers covering the same spatial extent. The spatial extent is the only common factor linking these layers, similarly instances of entity-sets A and C are independent of each other in all respects other than that they may be related to common instances of entity-set B." (Firns, 1994b:99)

Extending the analogy, Firns indicates that if thematic layers are appropriately defined, then only relationships within layers (i.e. intra-layer relationships) should be included in a conceptual schema. The necessary inter-layer relationships in a spatial database application would be easily derived using common GIS operations (overlay, point-in-polygon etc.).

4.3.2 Spatial Relationships Requirements

The second category of spatial concepts is the representation and the semantics of spatial relationships. Firns recognises the need for explicitly representing spatial relationships, as well as distinguishing between semantically different spatial relationships. In order to define distinct formalisms for these, Firns investigates the general nature of spatial relationships from the perspective of conceptual data modelling.

4.3.2.1 Hierarchical Order of Spatial Relationships

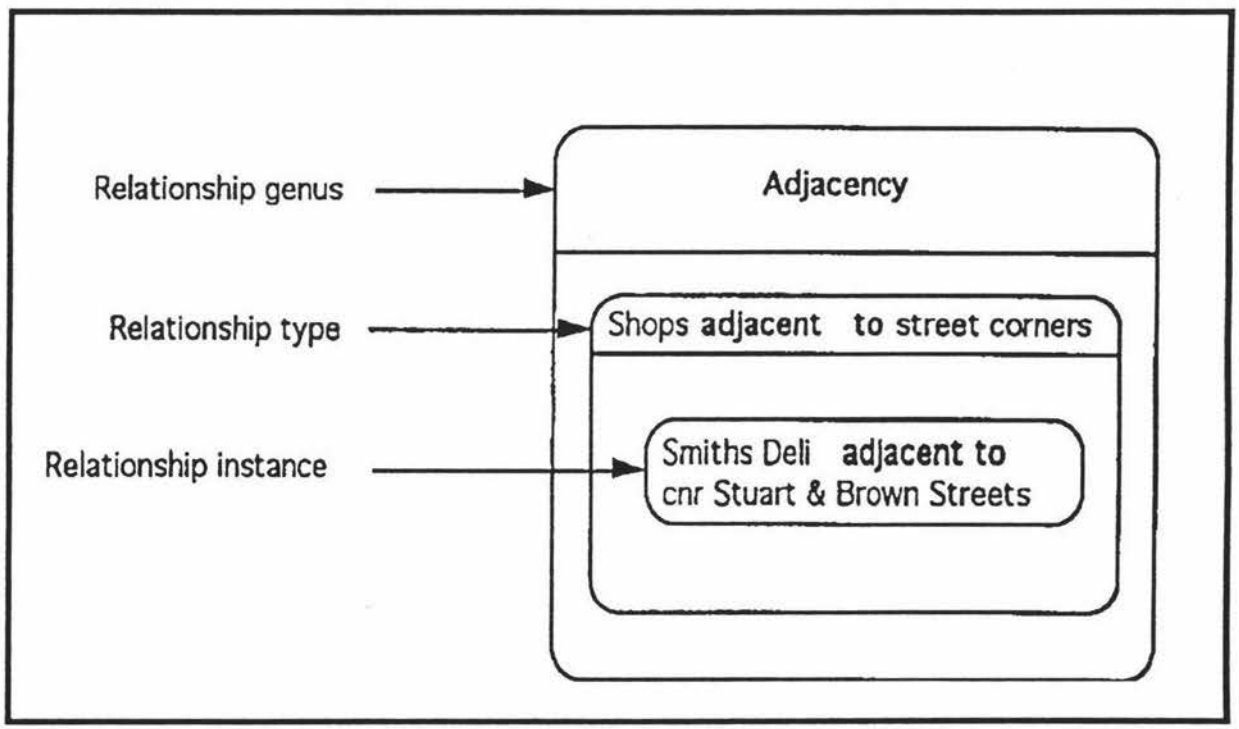


Figure 4.2 A Hierarchical Order of Relationships (Firns, 1994b:105)

The hierarchical order is illustrated in figure 4.2, the three orders being spatial relationship **genuses**, spatial relationship **types**, and spatial relationship **instances**.

Firns states, that any relationship, which is to be represented in a conceptual schema must be definable at each order of the above hierarchy.

Figure 4.2 shows a shop which is adjacent to a street corner, however, this does not mean that all shops and street corners are inherently associated with each other. Consequently, it would not be appropriate to define this relationship at the **type** level.

Firns identifies characteristics relevant to each order of the above hierarchy - these criteria must be meet for relationships to be represented at the conceptual data modelling level. These are:

" • *the relationships must be of such a nature as to be knowable or evident prior to he implementation of a database;*
• *the relationships must be relevant to the intended application(s) of the database;*
• *the relationships should be expected to be reasonably static;*
• *the relationships should be inherently structural or functional in nature."* (Firns, 1994b:106)

The last criterion warrants some explanation. Structural spatial relationships refer to those relationships which in the terminology of semantic data modelling are named as **Isa** or **Is-part-of** relationships. Functional, in this context, refers to whether an object is, in some way other than structurally, inherently associated with another object.

4.3.2.2 Classification of Spatial Relationships

Firns classifies the intra-layer spatial relationship genuses, those relationships which occur between objects within a thematic layer, as topological (Pullar and Egenhofer, 1988 cited by Firns 1994a; Egenhofer and Franzona, 1991) and structural (spatial Is-part-of) relationships. The spatial Is-part-of relationships are Firns' new concept and require some explanation.

Conceptual data models traditionally represent relationships between sets of objects with no distinction between spatial and non-spatial relationships. In Firns' opinion, in the context of SIS it is possible to classify relationships into three categories:

(a) those with no spatial connotations
(b) those which have both spatial and descriptive dimensions
(c) those which are purely spatial in nature.

Firns strives not to represent the third category of relationships with the help of the traditional abstraction mechanisms, for this practice could lead to ambiguities. That is why the spatial grouping and spatial aggregation relationships are proposed. These spatial relationships are analogous to the grouping and aggregation abstractions respectively and the terms are defined as follows.

"A spatial grouping is defined as an Is-part-of relationship, in that instances of one object type are defined as a grouping of all instances of another object type within a specified spatial extent." (Firns, 1994b:123)

A forest for example, comprises all the trees within its boundaries. The forest is not merely the defined region, it is the trees within the defined region and it could be defined as a spatial grouping of those trees, according to Firns.

"A spatial aggregation is defined as an object type which comprises an instance of each of two or more other (not necessarily distinct) object types...the spatial aggregation is determined on the basis that instances of the participating object types fall within a specified spatial extent." (Firns, 1994b:123-124)

The example given is a sport venue. A sport venue might be defined as a spatial aggregation of a playing field and a pavilion.

It could be debated whether or not Firns' above presented classification of spatial relationships is valid (i.e. whether or not the spatial Is-part-of relationship is a well-grounded extension to the standard ER model). Again, this thesis is going to apply the data modelling objectives to think about the above question. The two, sometimes conflicting objectives of conceptual data modelling are to model reality and to form the basis for logical database schema design (Firns, 1990a). It could be stated that these abstraction mechanisms burden designers in their attempt to represent reality, i.e. the identification of spatial structural relationships requires refined analysis. If the above classification has any benefits in respect to the second objectives, the classification is not superfluous. However, benefits could not be discovered in the case of relational schema derivation (Firns, 1994b:165-166). This one-sided, simple evaluation is not intended to negate Firns' results. Here, only a question is asked, namely, is adding the spatial Is-part-of relationships to the present state of knowledge really necessary?

4.3.3 Summary of the Required Abstraction Mechanisms

Figure 4.3 summarises the abstraction mechanisms supported by the new conceptual model. "Spatial Coincidence" is shown in the figure, but it is not an abstraction mechanism, and is not to be modelled in conceptual schemas.

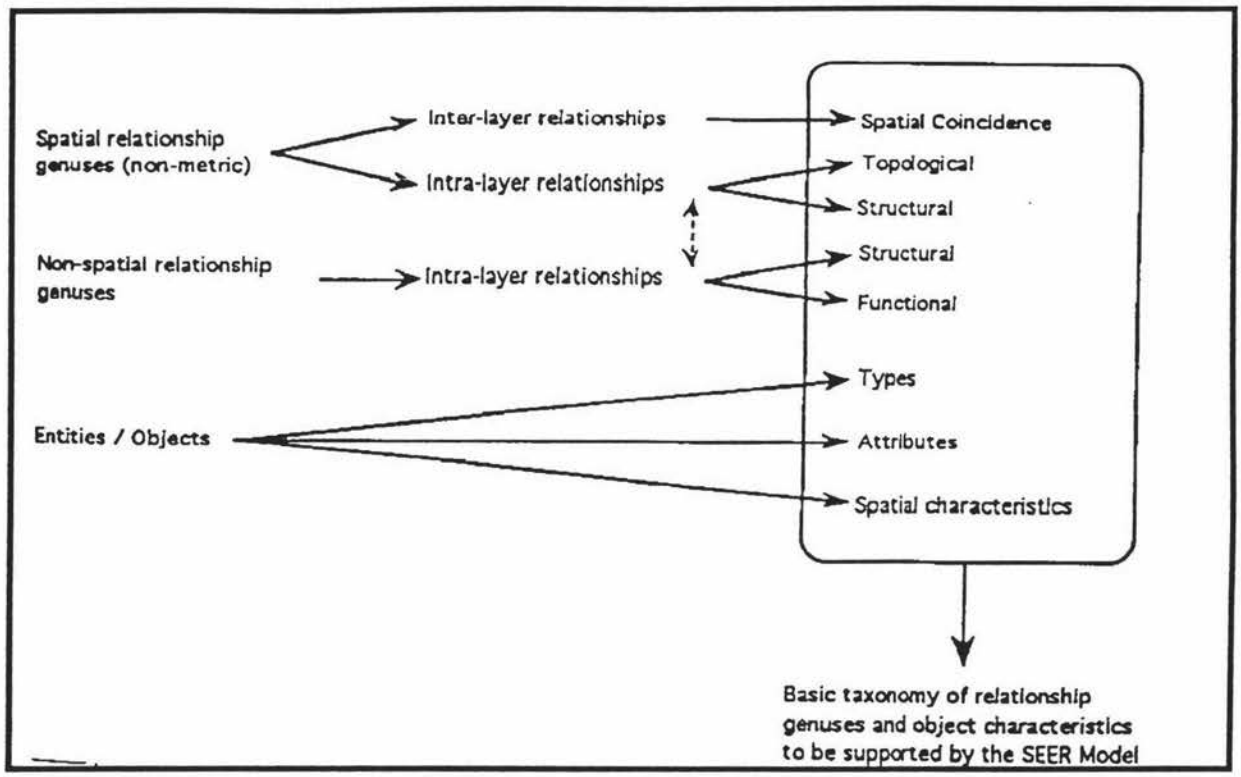


Figure 4.3 General Taxonomy of Relationship Genuses and Entity Characteristics to be Supported by the SEER Model (Firns 1994b:112)

4.4 A Spatially Extended Entity Relationship Model

The spatially extended entity relationship (SEER) model provides for structural modelling (Brodie, 1984). The SEER model has three components corresponding to the non-spatial and spatial categories of requirements (refer to 4.3). The components are: the standard entity-relationship model, the locational data modelling component and the spatial sub-model. These are shown in figure 4.4.

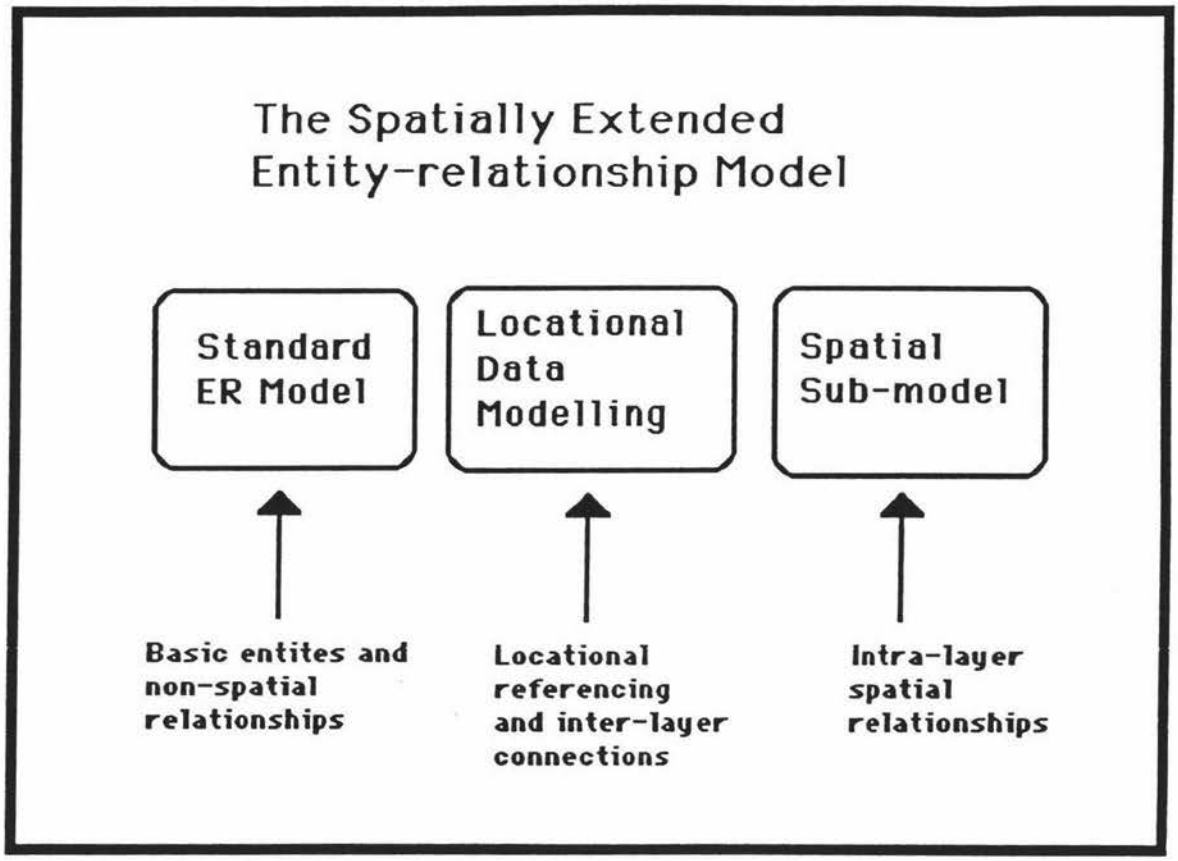


Figure 4.4 Three Components of the SEER Model (Firns, 1994b:131)

The major components encompass abstraction mechanisms which enable the differentiated modelling of the spatial and non-spatial aspects of the problem domain. This section examines the components of the SEER model.

4.4.1 Entity-Relationship Model in the SEER Model

The Finkelstein (1989) version of the ER model is used to model basic entities and non-spatial relationships.

The notation presented by Finkelstein is chosen for a number of reasons. Firns acknowledges as valid the critique that this diagramming notation lacks information about business rules. He argues that the simplicity of the notation is an advantage in this context. The notation does not over-shadow the real problem at hand - modelling spatial concepts. Another advantage of the notation is that it conveys the same data semantics in a more compact manner than the usage of Chen's notation (Firns, 1990a).

The diagramming convention is illustrated in figure 4.5. The entity sets are depicted by rectangular boxes and relationships by lines connecting pairs of entities. The basic differences between the notation adopted by the SEER model and Chen's extended

version are that the relationship sets are not named, and attributes of entity sets or relationship sets are not shown in the diagram.

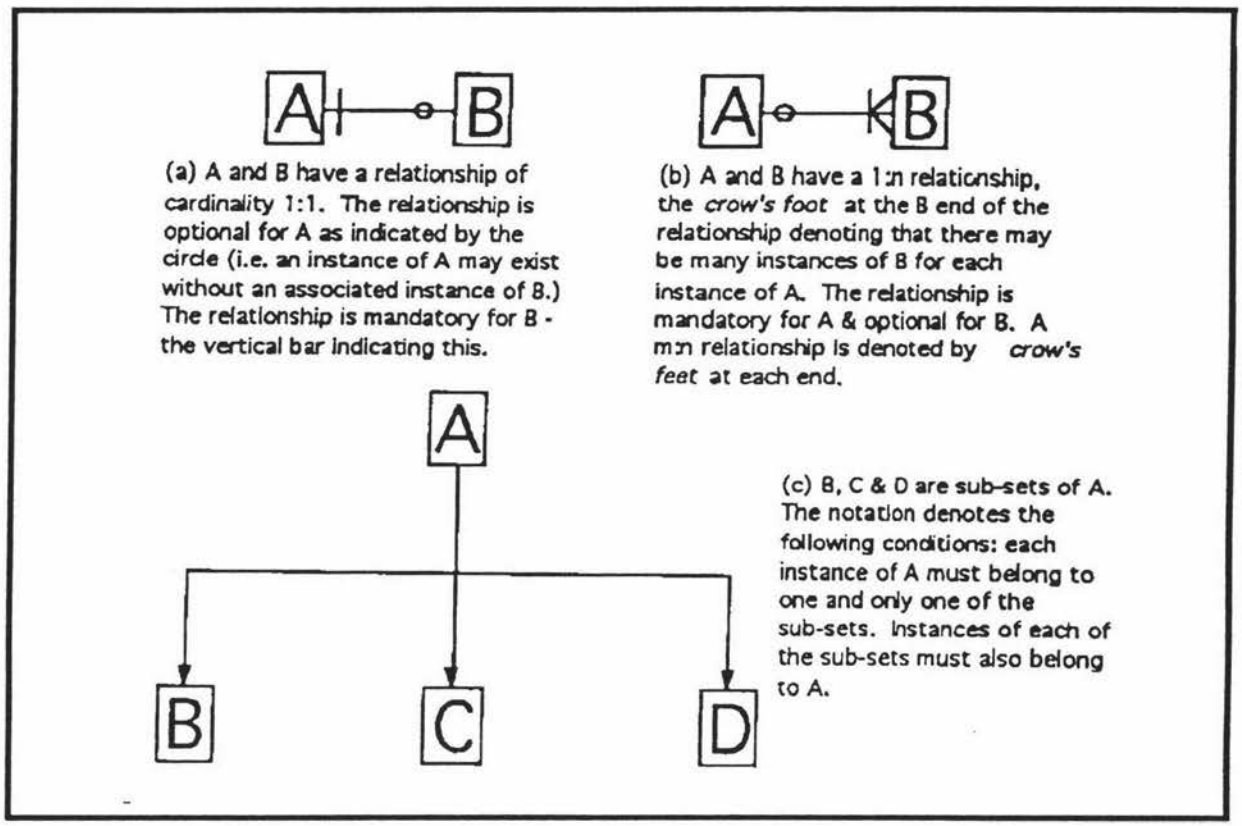


Figure 4.5 ER Diagramming Notation (Firns, 1993:10)

4.4.2 Locational Data Modelling in the SEER Model

The key feature of the second, locational data modelling component is deduced from the analogy between thematic layers and parallel linkages.

There are a number of problems which have been encountered during the interpretation of this part of the dissertation (Firns, 1994b:133 -136). The first problem emerged with the following introduction of the locational data modelling components.

*"...two special types of entity sets with extended semantics, the LOCATION entity set and **node** entity sets, are used as the basis to model reality as comprising a number of thematic layers."* (Firns, 1994b:134)

No clear, precise definition has been found for the entity set LOCATION, although Firns defines characteristics of the LOCATION entity set with the expression *"by definition"* (Firns, 1994b:134-135). The comments that are presented in section 4.3.1.1

of this thesis, *"represent location"* (1994b:89) and *"basis to model location"* (1994b:134) give an indication of two possible interpretations. The first interpretation is that the LOCATION entity set represents the **spatial extent** to be modelled (i.e. B in figure 4.1 (iii)). What is the spatial extent? **Space,** as defined by the Concise Oxford Dictionary, is *"a continuous extension viewed with or without reference to the existence of an object within it"*. **Extent** is defined, by the same dictionary, as *"space over which a thing extends"*. **Spatial extent** is interpreted in this thesis as the space over which the real world objects, which are to be modelled, extend.

The second interpretation of the LOCATION entity set is that it represents the sets of all spatially referenced objects, which have one or more roles (themes or nodes) in the overall database.

Choosing the name **node** for the second special entity set is not ideal as it offers unintended connotations in a spatial context. (i.e. node is a spatial object which is often used to indicate a point in 2 dimensions.) The definition is given for the node sets: *"The **node** entity sets are the means by which thematic layers are represented in SEER schemas."* (Firns, 1994b:135)

The following is given as definitions of the inter-layer and intra-layer relationships:

"The relationships between LOCATION and all node entity sets are the basis for the representation of inter-layer connections in SEER schemas. Intra-layer relationships are derived from the semantics of sub-types of node entity sets and relationships between these and the usual entity sets of entity relationship model." (Firns, 1994b:134)

For a discussion of the spatial representation class attribute refer to section 4.2. The geometric data type or geometric class attribute is discussed in section 4.3.1.2.

Table 4.1 gives a summary of the terms as they are interpreted in this thesis.

• LOCATION entity set	• The basis to model location in the SEER model. • The role of LOCATION is to model the underlying spatial coordinate system which will be used to implement the model. • An instance of the SEER model incorporates only one such entity set. • LOCATION does not have other instance attributes apart from the key attribute. • LOCATION is only related to node entity sets, relationships to node entity sets are of cardinality 1:1 or 1:n
• node entity sets	• The means by which thematic layers are represented - any spatially distributed phenomena not directly related to the theme, but associated with the location requires a new node entity set definition • any two entity sets related to the same node sub-type must be spatially related to each other • A node entity set is any entity set which participates in a 1:1 or 1:n relationship with LOCATION, such that any instance of the entity set can be related to only one instance of LOCATION.
• inter layer relationships	• The relationships between LOCATION and all node entity sets are the basis for the representation of inter-layer connections in SEER schemas.
• the spatial representation class attribute of node entity sets	• The value of which specifies the spatial representation model by which the associated map layer will be implemented - for example raster or vector. It is the connection to the representation layer.
• the geometric class attribute[2] of entity sets related to a sub-type of a vector typed node entity set (Such an entity set is referred to as a locationally referenced entity set.)	• Denotes the way in which instances of locationally referenced entity sets will be geometrically represented. The value of the geometric class attribute imposes constraints on the way in which entity sets can be topologically related in the separated spatial sub-model component of a SEER schema.

Table 4.1 Basic Terms for Locational Data Modelling in the SEER model

Diagramming notation for the basic terms is shown in figure 4.6.

2 Also called "geometric data type"

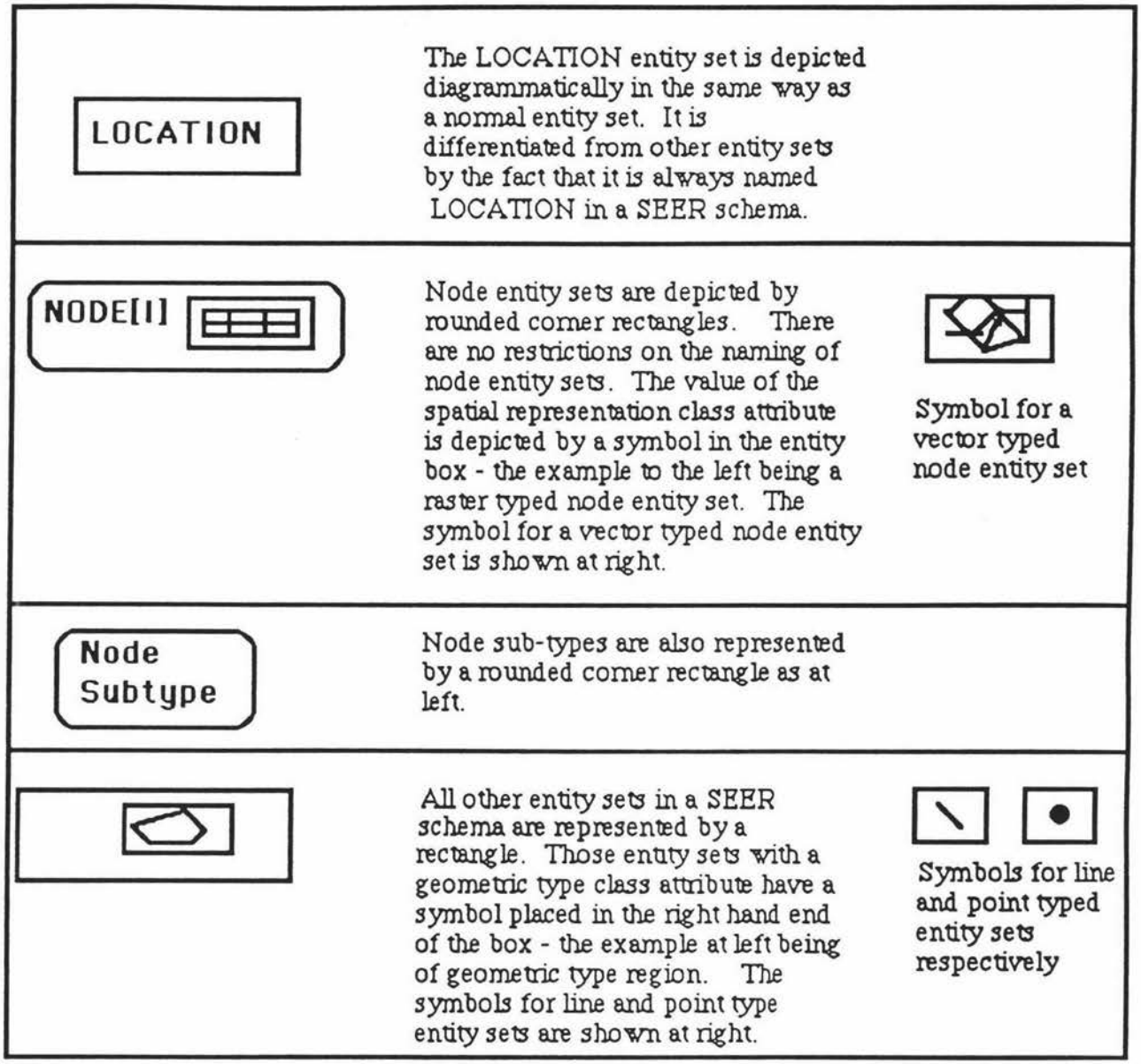


Figure 4.6 SEER Diagramming Notation for Locational Data Modelling (Firns, 1994a:293)

4.4.3 The Spatial Sub-model of the SEER Model

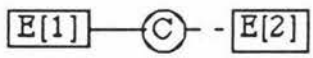
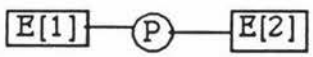
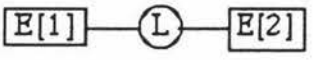
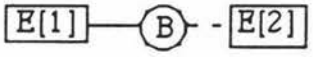
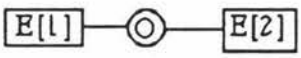
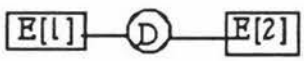
The third major component of the SEER model, the spatial sub-model is only applicable to modelling spatial relationships in vector layers. *"This is because a raster layer has no underlying spatial structure, other than an arbitrary division of space into cells."* (Firns, 1994b:145)

The sub-model is an independent model from the previous two components, although the value of the geometric class attribute of a vector typed locationally referenced entity set in the locational modelling component imposes constraints on the way in which entity sets can be topological related in the spatial sub-model component of a SEER

schema (i.e. a point typed entity could not 'contain' another entity). The sub-model, strictly speaking, does not model the real world.

Spatial relationships genuses within either of the categories (topological and Is-part-of) are mutually exclusive for any pair of entity instances, but a pair of entity sets may participate in a topological and an is-part-of spatial relationship.

Figure 4.7 and figure 4.8 describe the topological and Is-part-of spatial relationship genuses respectively.

Spatial Relationship Genus	SEER Notation	Valid geometric types for entity sets E1 and E2 (E1/E2)	Description & Comments
contain		region/region * region/line region/point line/line * line/point	Instances of entity set E2 are each contained within an instance of entity set E1. E1 and E2 are not necessarily distinct.
point adjacent		region/region * region/line region/point line/line * line/point point/point *	Instances of entity sets E1 and E2 have a point in common. E1 and E2 are not necessarily distinct.
line adjacent		region/region *	Pairs of instances of E1 and E2 have common line segments. E1 and E2 are not necessarily distinct.
boundary		line/region point/line	Instances of E1 form parts of the boundaries of instances of E2.
overlap		region/region * line/region line/line * point/region	Instances of entity sets E1 and E2 may overlap E1 and E2 are not necessarily distinct.
disjoint		region/region region/line region/point line/line line/point point/point	Instances of E1 and E2 are disjoint - i.e. they are not topologically related to each other. E1 and E2 are not necessarily distinct.

* For those pairs of geometric entity types in the third column which are marked with an asterisk, E1 and E2 need not necessarily be distinct.

Participation for entity sets in relationship types of any of the above relationship genuses, except disjoint, may be specified as optional. The same notation as for optional participation in spatial groupings and spatial aggregations is used - i.e. a circle placed on the connecting line.

Cardinality may also be specified for the topological relationships (other than disjoint relationships). The same notation as for specifying cardinality in the Finkelstein version of the entity relationship model (i.e. the 'crows foot') is used in the SEER model.

Figure 4.7 Topological Spatial Relationship Genuses in the SEER Model (Firns, 1994a:294)

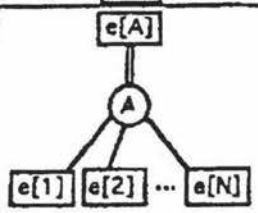
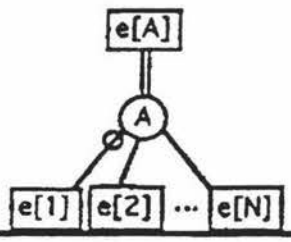
Spatial Relationship Genus	SEER Notation	Description
spatial grouping		Each instance of a spatial grouping type comprises a finite set of instances of entity set E^1.
spatial aggregation.		Instances of a spatial aggregation type, E^A, each comprise one instance of each of entity sets E^1 to E^N.
Participation of entity set E^1 in a spatial grouping, or any of entity sets E^1 to E^N in a spatial aggregation can be specified as optional. The notation used to indicate optional participation is a circle placed on the connecting line for the appropriate entity set. The example at right shows optional participation for entity set e^1 in a spatial aggregation.		

Figure 4.8 Is-part-of Relationship Genuses in the SEER Model (Firns, 1994a:294)

4.5 Data Modelling Objectives and the SEER Model

The SEER model sets out to fulfil both data modelling objectives.

The first data modelling objective is to model reality. The proposed model strives to support the representation of real world objects, spatial and non-spatial characteristics of these objects, and the spatial and non-spatial relationships between them.

The standard ER model and the locational data modelling components model the real world. Firns, with the incorporation of the spatial representation class attribute, bridges the gap between the real world objects and their geometric representation. The third component, the spatial sub-model, does not, strictly speaking, represent the real world as it is defined in this thesis (refer to chapter 2.4), but this component is separated from the real world representation.

The second objective of conceptual data modelling is to form a base for the logical database design. Although the SEER model is regarded by Firns as being in an

"experimental" stage, he demonstrates its practical usefulness by specifying relational schema derivation rules applicable to a SEER schema (Firns, 1994b:152-177).
Firns also presents a comparison between the standard entity relationship and the spatially extended relationship model by comparing schemas developed for the same database application. The comparison shows benefits of the SEER model while maintaining that *"These advantages in no way diminish the value of the entity relationship model as a conceptual model - the extended notation of the SEER model complements that of previously existing versions of the entity relationship model"* (Firns, 1994b:177).

From the perspective of this thesis two advantages of the SEER model are important.

First, its independence from the implementation level spatial data model[3]. This is an advantage, because the conceptual schema of chapter 5 could be independently produced from the specific spatial data model of the TECHBASE[4] software. While using the generic entity relationship model it is not feasible to model spatial semantics if the intended implementation level spatial data model is not known.

The second important advantage is that the SEER model explicitly models spatial semantics, while the entity relationship model uses no distinctive notation for spatial and non-spatial semantics. Therefore the resulting ER schema would incorporate a model of reality and a particular geometric model of reality.

It is felt that in addition to the previously mentioned advantages, the SEER model comprises many new valuable concepts, worthy of close attention even in this "experimental" stage. This is why the SEER model will be applied to the case study.

---

[3]The fact that the SEER model defines specific geometric data types means that these must be supported by the logical level model, therefore the SEER model and the logical level spatial data model are not completely independent of each other. The form of logical model is otherwise independent of the SEER schema (Firns, 1994b:153).

[4]Refer to Appendix D section 4.4

4.6 Chapter Summary

This chapter introduced a recently proposed data model which is applicable to spatial database design. The SEER model is a conceptual data model in the taxonomy of spatial data models adopted in this thesis, the only one of this kind encountered during the author's research.

The SEER model encompasses numerous new concepts for the representation of the spatial and non-spatial semantics of the database application. These new concepts were described and some criticisms of them were included. Reasons for selecting the SEER model for the case study were also given.

Chapter 5

Case Study - Modelling for an Underground Water Database

5.1 Introduction

This chapter develops a conceptual schema of the Manawatu-Wanganui Regional Council (MWRC) Groundwater Section database by applying the spatially extended entity relationship model (Firns, 1994a, 1994b).

The conceptual database design presented in this chapter is based on the data requirements collection exercise carried out at the headquarters of the Manawatu-Wanganui Regional Council in June 1994, and is documented in the Appendices of this thesis. Appendix A contains the preparation of the data requirements collection and the questionnaire created for that purpose. Appendix B is a general introduction to the Manawatu-Wanganui Regional Council, Appendix C shows organisation charts of the MWRC and Resource Monitoring Department. Appendix D contains a report which sums up the experiment at MWRC. The report introduces the groundwater management responsibilities, reviews the sources of the section's data and gives a background to the history and the present state of the Groundwater Section database. TECHBASE software, used by the section is described. Present data requirements and data deficiencies of the section's activities are identified. In Appendix E the database definition of the presently functioning database is enclosed.

This chapter deliberately overlooks practical difficulties and data acquisition expenses when identifying the main layers of the future database. The entities of the layers are described, main business rules are inferred, and graphical conceptual schemas are presented.

5.2 Thematic layers : Inter-Layer Relationships

The thematic layers for the future database are identified based on the data requirements and the data deficiencies of the underground water database, documented in Appendix D (refer to sections D.6 and D.7). The thematic layers are:

- Groundwater Node
- Catchment Node
- Road Node
- Soil Node
- River Node
- Land_Use Node

The nodes above constitute the SEER schema shown in figure 5.1.
Node entity sets of the "context diagram" represent thematic layers in the future database. Figure 5.1 can be interpreted as: any location (spatial extent) recorded in the database can have data kept in one of the identified thematic layers. Data kept in thematic layers must belong to only one location.

The spatial representation class attribute of the node entity sets have the value "vector" because the software used by the groundwater section implements the vector model to store spatial data (refer to Appendix D, section D.5.1).

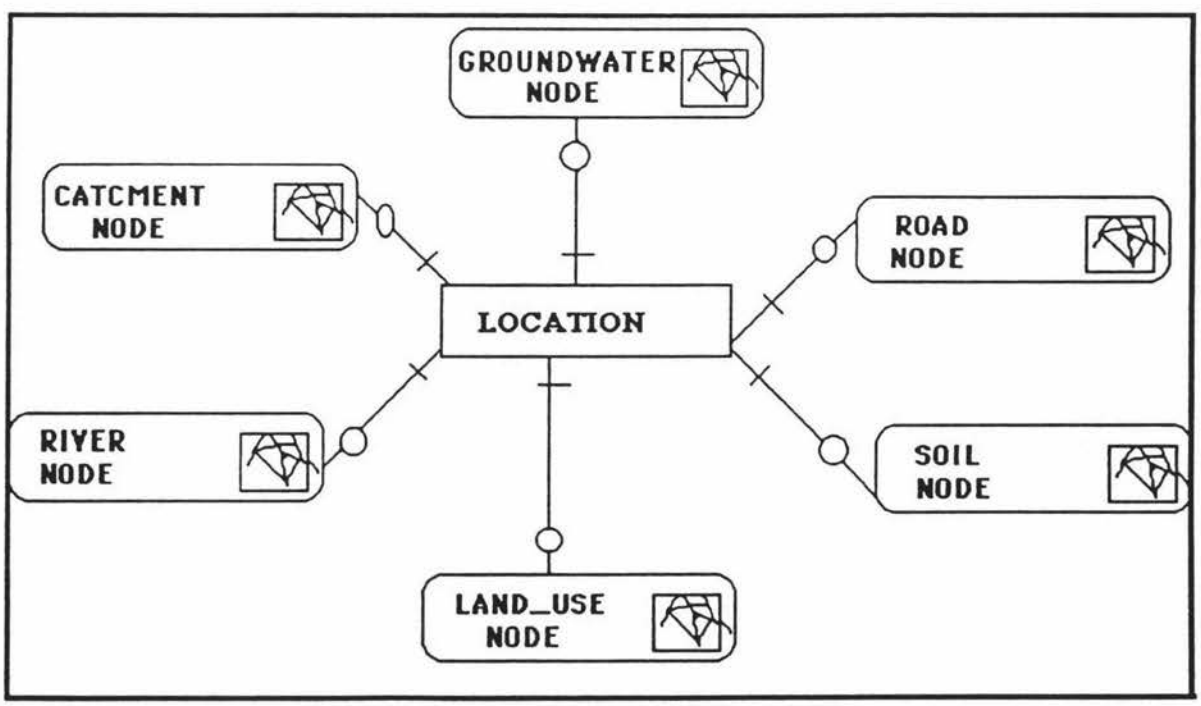


Figure 5.1 "Context "Diagram for the Underground Water Database

Data belonging to two of the above thematic layers exist in the present database i.e. the Groundwater and Catchment Node thematic layers.

• The Groundwater Node stores most of the data presently used by the section.

• The Catchment Node also exists presently. The region has been divided into groundwater catchments or areas for statistical and groundwater usage restriction data purposes by the underground water scientist. The names of the areas are: Wanganui, Whangaehu-Turakina, Rangitikei, Manawatu, Horowhenua and Eastern area.

• The Land Use Node, Road Node and River Node are needed for data reference purposes. Clients refer to the location of bores with respect to the nearest roads, streets, major fence lines, drains or water courses which can be readily seen on maps. The digitised road, cadastral and hydrographic data would enable on-screen searches (refer to Appendix D, section D.7.3).

• The Soil Node stores soil type data, which is needed for two reasons. Soil type influences the rate of groundwater recharge. i.e. how much of the rainfall seeps through the aquifer, and soil type is an important factor when evaluating aquifer vulnerability contamination from the surface.

5.3 Intra-Layer Entities and Relationships

This section describes the entities and relationships of each thematic layer. The designation of primary key attributes for each entity set is included, which enables relationships between entities to be represented by foreign keys. The lists of attributes are kept to a minimum throughout this chapter for two reasons. The main attributes are listed in Appendix B and all attributes are defined in Appendix E. References to these appendices seemed sufficient from the perspective of this exercise. The second reason for not repeating the attributes in full length is the time constraint of this thesis.

5.3.1 Groundwater Layer - Entity Sets

The Groundwater Layer is shown in figure 5.2. The solid arrow notation between the Groundwater Node (node entity set) and its sub-types (BORE-NODE and PROPERTY-NODE) is an adaptation of a diagramming technique used to represent entity sub-types in an extended ER model (Ferguson, 1988 cited by Firns, 1990:17; refer to figure 4.5).

The main entity sets are described as follows.

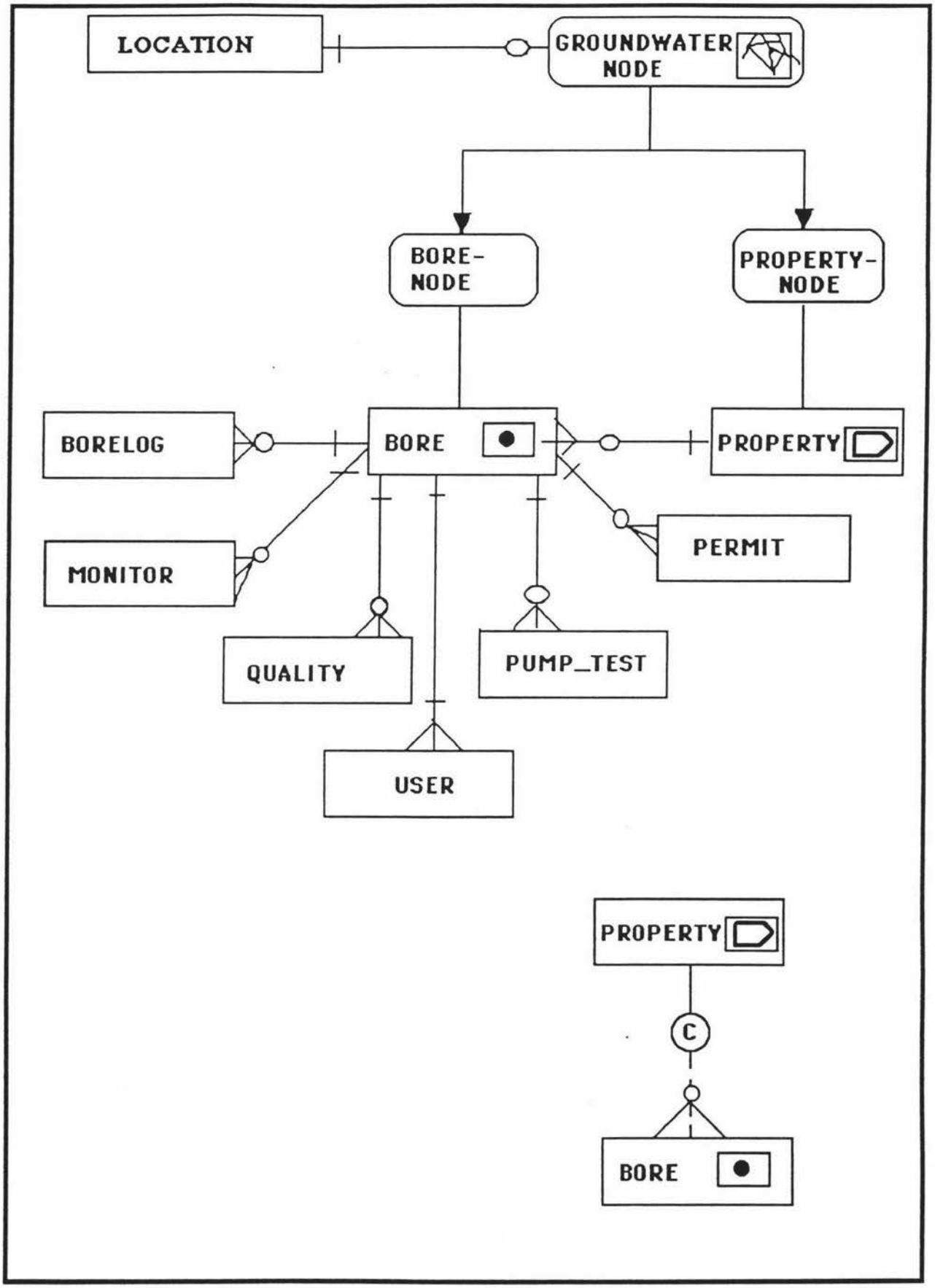


Figure 5.2 Groundwater Layer

BORE

Bores play an important role in groundwater management because groundwater resources are only accessible through springs, bores or wells. Groundwater is extracted from aquifers via bores. Bores provide the means for groundwater data acquisition and the monitoring of underground water levels, sampling and analysis is only possible by irregularly spaced bores. Bores serve as control points for groundwater modelling.

If the bore is deepened[1], groundwater level, chemistry of the water, transmissivity and other parameters can change. For this reason, if a bore is deepened, it is issued with a new and unique holeid number. Information from the old and new, deepened bore are regarded as information from two different boreholes; the location (easting and northing) of the two bores are the same.

If a bore is backfilled or filled up with cement, it ceases functioning. However, information from this bore is still valuable and should be recorded. (e.g. 400 m deep bore is drilled, water quality is tested and found to be inferior. The borehole is filled up with cement; the information that there is bad quality water at 400 m is still valuable.)

In the existing database each bore has a unique identifier, holeid, consisting of grid and well_no fields (refer to Appendix D, section D.4.2). Bores are described by their location on the surface(easting and northing), their depth and numerous non-spatial, descriptive data.

Bore is a locationally referenced, point typed entity set.
The main attributes for each bore are:

- holeid (surrogate key)
- depth
- usage
- drilling details (name, date)
- status - a new attribute to indicate if the bore is backfilled.

[1]The depth of the bore is the distance between the bore head (on the surface) and the place where water enters the bore.

PROPERTY

This is a locationally referenced, region (or in TECHBASE software terminology, polygon) typed entity set.

The owner(s) of a property owns the bore situated within the property. A property can contain many bores, there are properties with more than ten bores. The most important attributes are:

- property_Id (primary key)
- owner name
- owner address.

USER

The user(s) of the bore can be different to the current owner(s) or it is possible, that there are many users of the same bore. The owner and/or principal user may decide to supply groundwater to the neighbour(s) for example, or it may be a rental property. The most important attributes are:

- holeid + user name (primary key)
- user address.

USER entity set is a weak entity set (refer to chapter 3, table 3.1), it uses BORE primary key as part of its primary key.

PUMP_TEST

Pump tests are described in Appendix D, section D.2.2, the data from pump tests is stored in the table LOCATION (refer to section D.5.2).

Pump tests may be conducted many times at a given bore. The main attributes are:

- holeid + date (primary key)
- static water level
- discharge
- drawdown
- specific discharge
- coefficient of storage
- aquifer transmissivity.

MONITOR

Groundwater levels are monitored by the MWRC at selected bores within the region at regular time intervals. Monitor attributes are the same as the attributes in table WOBS-groundwater level and flow observations (Appendix D, section D.5.2). The main attributes are:

- holeid + date (primary key)
- observed static water level
- flow.

QUALITY

Groundwater quality data is the measurement data of various physical, chemical and biological constituents. Groundwater quality data is time dependent and monitored at selected bores only. Quality attributes are the same as the attributes in table QUALITY (Appendix D, section D.5.2). The main attribute is:

- holeid + date (primary key).

PERMIT

Permits are required for the drawing of a maximum amount of groundwater, at a given location(s) from a certain depth(s) and are for a limited time period. Once that period has elapsed, the permit needs to renewed. The renewed permit may differ from the previous one. Consequently, several permits (issued at different times) can refer to the same bore. Many bores have no permits at all. The main attributes are:

- permit number (primary key)
- time period
- maximum amount of groundwater.

BORELOG

A bore may have one or more "borelog" records.

Borelog attributes are the same as the attributes in the table LITHO (Appendix D, section D.5.2). The main attribute is:

- holeid + LogNumber (primary key).

5.3.1.1 Business Rules for the Groundwater Node

Although the diagramming notation used by Finkelstein and adopted by the SEER model does not give information on business rules (refer to chapter 4.), in this thesis the relationships between entities of the Groundwater Node are named and business rules are inferred.

Business rules can be inferred from relationship names, cardinality and participation constraints, to facilitate comprehension of the conceptual schema. Business rules for the Groundwater Node (refer to figure 5.3) are shown in table 5.1.

Cardinality	Entity	Parti-cipation	Relation-ship	Cardinality	Entity
One	BORE	can	have	one or more	BORELOG
One	BORELOG	must	pertain to	only one	BORE
One	BORE	can	be monitored	one or more	MONITOR
One	MONITOR	must	be related to	only one	BORE
One	BORE	can	be observed	one or more	QUALITY
One	QUALITY	must	belong to	only one	BORE
One	BORE	must	have	one or more	USER
One	USER	must	use	only one	BORE
One	BORE	can	be tested	one or more	PUMP_ TEST
One	PUMP_ TEST	must	test	only one	BORE
One	BORE	can	have	one or more	PERMIT
One	PERMIT	must	belong to	one	BORE
One	BORE	must	be situated	only one	PROPERTY
One	PROPERTY	can	contain	one or more	BORE

Table 5.1 Business Rules of the Groundwater Node

5.3.1.2 Spatial relationship for the Groundwater Node

Between the PROPERTY and BORE entity sets there is a spatial (topological) relationship - contain (refer to chapter 4, figure 4.7).

It could be debated whether or not including the PROPERTY entity set in the Groundwater Node is a valid design decision. It is more likely that in a "real life" design there would be a separate thematic layer designed for PROPERTY data. Instead of storing relationships between the PROPERTY and BORE entity sets, as it is suggested in figure 5.2, an overlay operation would be used to establish connection between the two entity sets stored in two distinct thematic layers.

5.3.2 Catchment Layer - Entity Sets

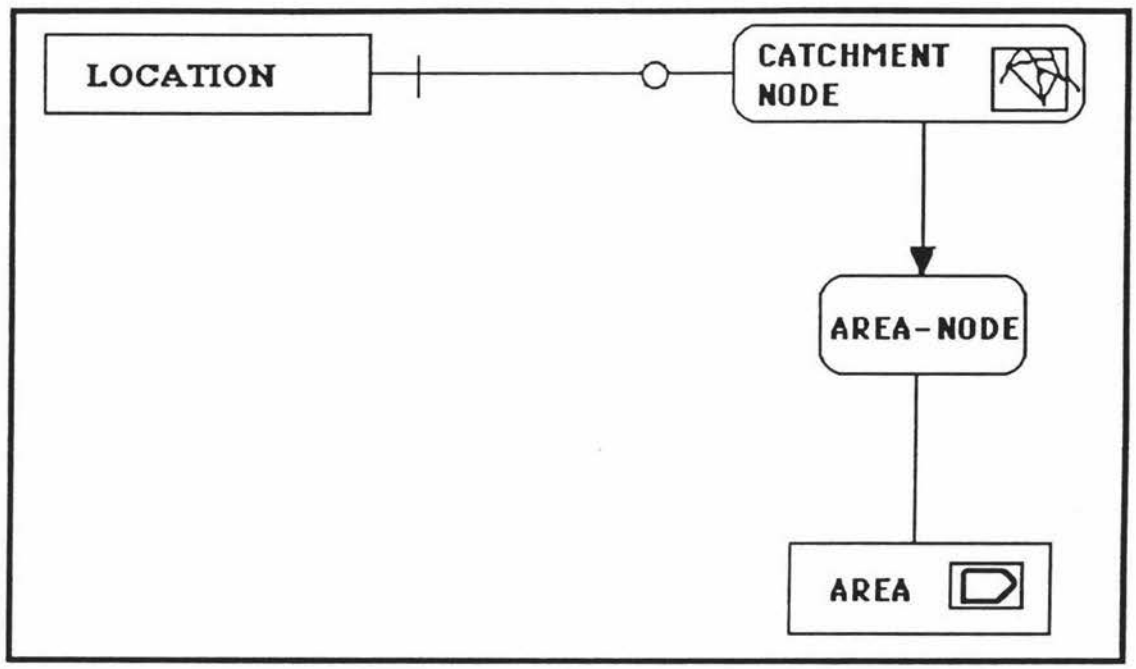


Figure 5.3 Catchment Layer

Only one locationally referenced entity set is identified for this layer - an AREA is a region (polygon) typed entity set. Statistical and groundwater usage restriction data are kept in this layer.

5.3.3 Soil Layer - Entity Sets

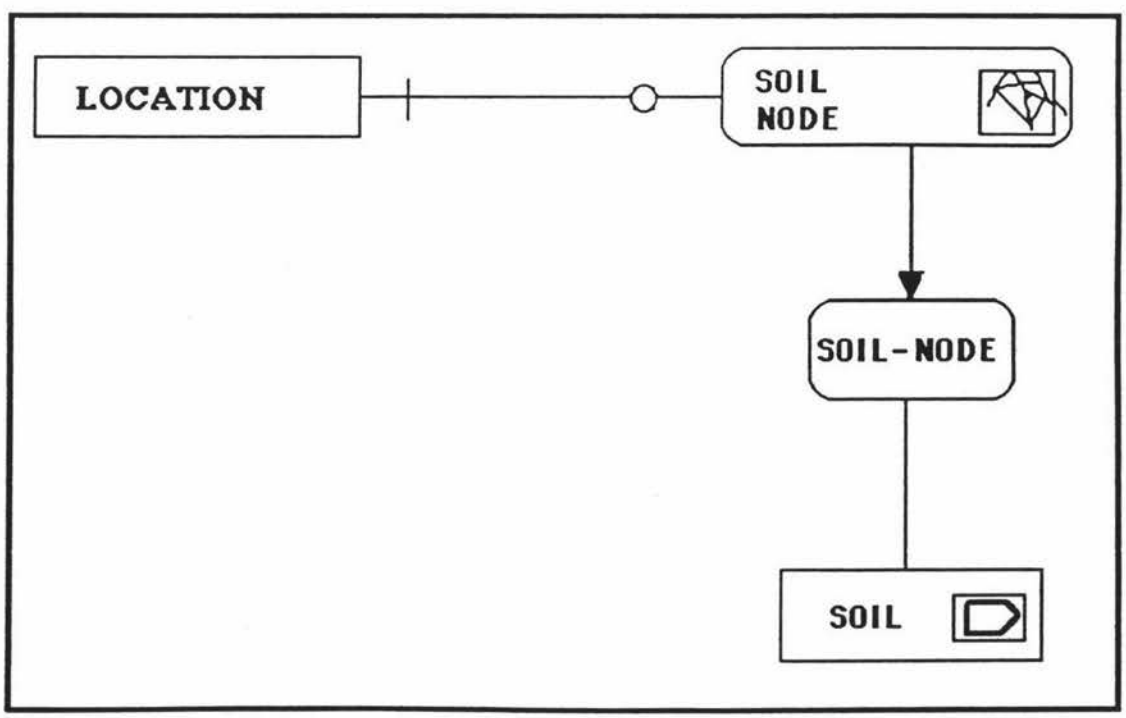


Figure 5.4 Soil Layer

Only one locationally referenced entity set is identified for this layer - a SOIL is a region (polygon) typed entity set.

5.3.4 Road Layer - Entity Sets

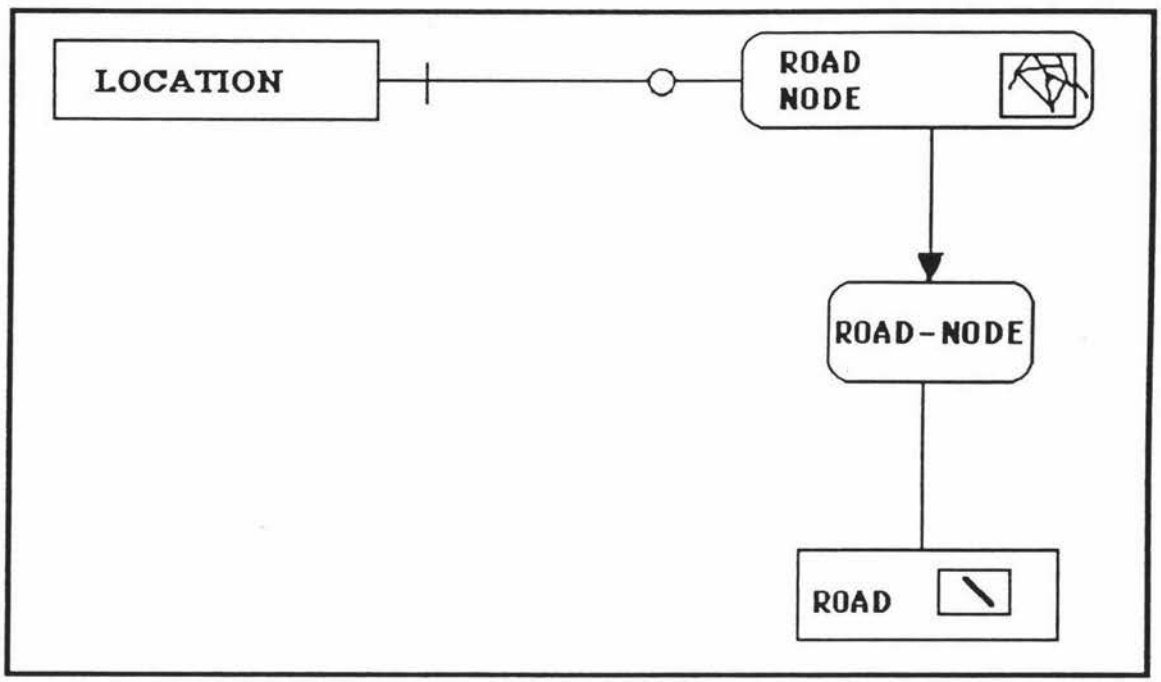


Figure 5.5 Road Layer

Only one locationally referenced entity set is identified for this layer - a ROAD is a line typed entity set.

5.3.5 River Layer - Entity Sets

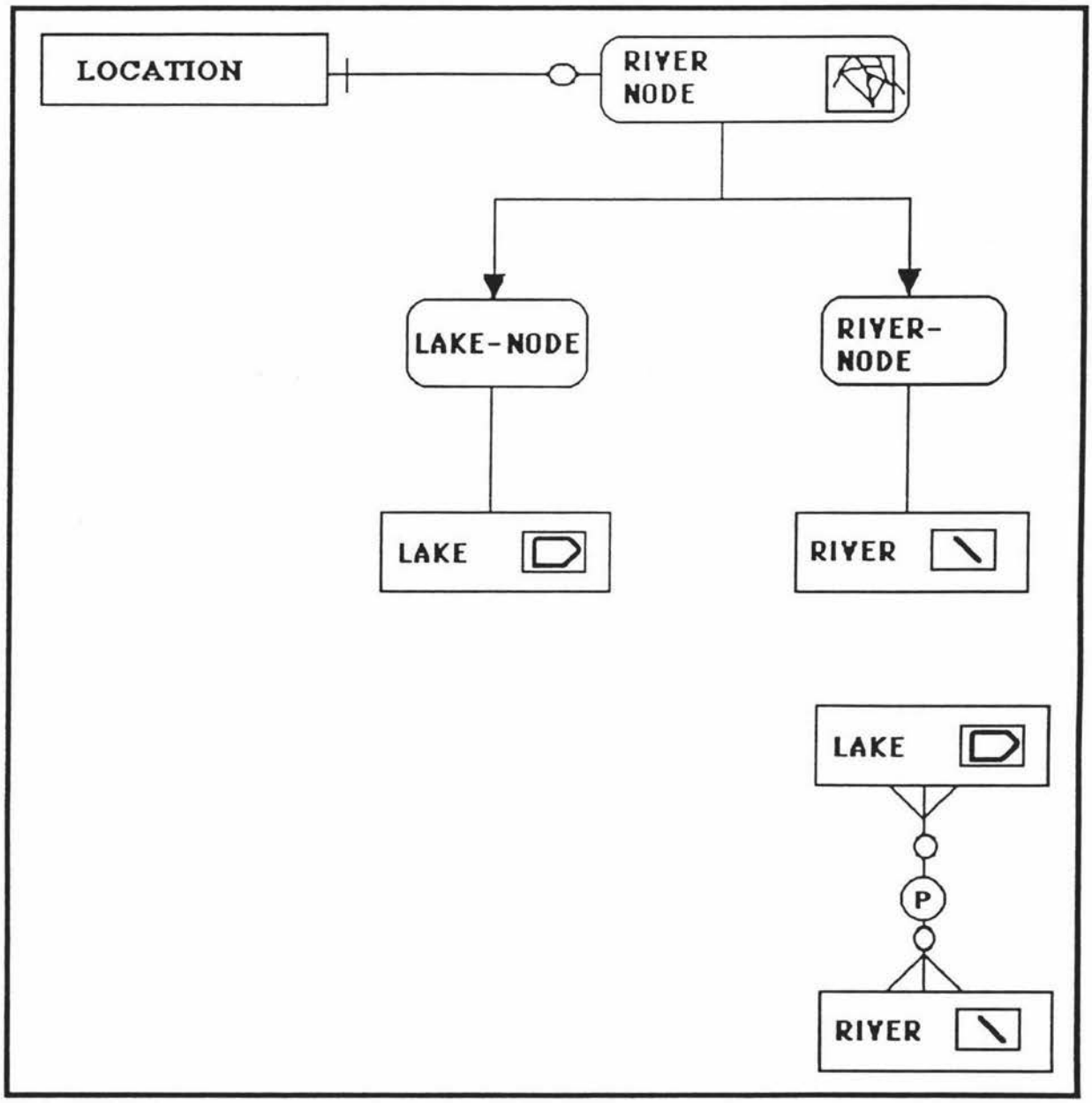


Figure 5.6 River Layer

Two locationally referenced entity sets are identified in this layer - a LAKE is a region (polygon) typed entity set and a RIVER is a line typed entity set.

The spatial sub-model identifies a spatial (topological) relationship between the LAKE and RIVER entity sets - a point adjacent relationship (refer to chapter 4, figure 4.7).

5.3.6 Land_Use Layer - Entity Sets

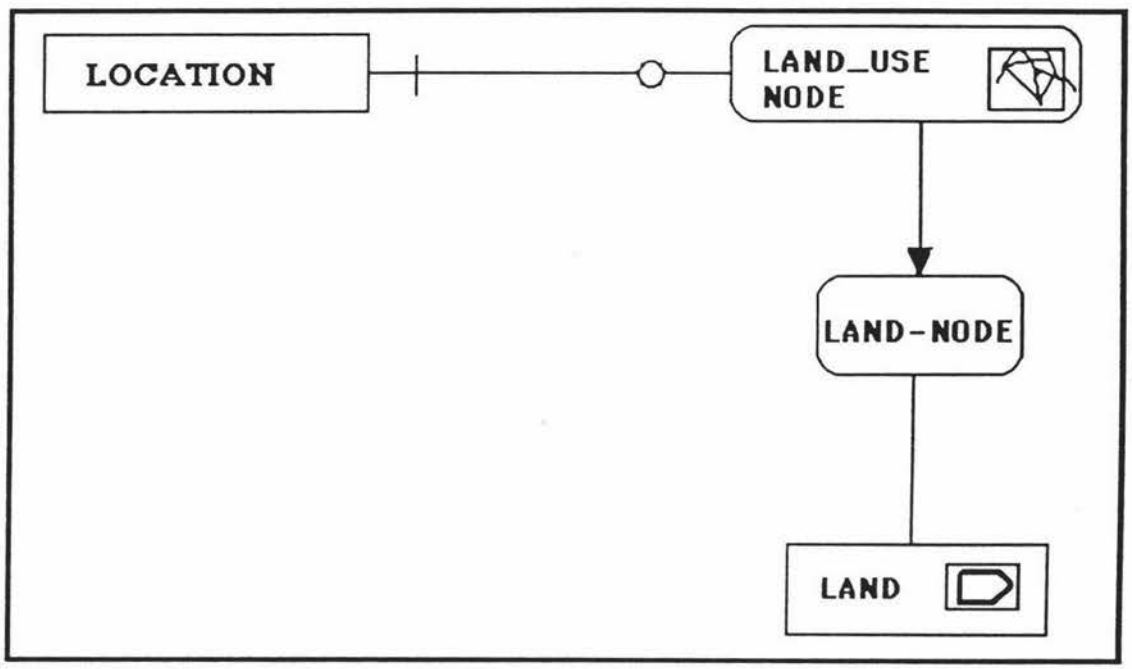


Figure 5.7 Land_Use Layer

Only one locationally referenced entity set is identified in this layer - LAND a region (polygon) typed entity set.

5.4 Findings

This small scale data modelling exercise shows some very important advantages of the SEER model.

Firstly, the components of the data model give clear guidelines for recognising and separating real world objects and modelling their geometric representation. The model helps to avoid the most general pitfalls of spatial conceptual data modelling (i.e. not distinguishing between the two different levels of perception).

Secondly, the SEER model is a conceptual data model, since it is independent from the implementation level spatial data model. The conceptual schemas were produced without considering the details of the implementation environment.

The conceptual schemas constructed in this chapter are feasible and could serve as the base for the logical database design. The logical schema could be produced and

implemented by adapting Firns' relational schema derivation rules (Firns, 1994b:152-177) for the structure in which spatial data is stored in the intended GIS software. (Presently, the Groundwater Section uses a relational database management system, extended with GIS capabilities. The derivation rules cited above would need minimal adjustments.)

Chapter 1 raises the question: does an appropriate tool (i.e. conceptual data model) exist for the practitioners of GIS database design? This chapter gives a positive answer to this question: the SEER model is an appropriate conceptual model in a spatial context.

5.5 Chapter Summary

This chapter constructed a high level view of a proposed database. Feasible conceptual schemas were presented for an underground water database. Thematic layers, main entities, spatial and non-spatial relationships were identified.

Although the time available for the database design exercise presented in this chapter was severely limited its conclusions are potentially useful for a database implementation or for future research activity in this domain and the main question of this thesis has been answered.

Chapter 6

Research Issues and Thesis Conclusion

6.1 Proposed Future Research

This thesis presents a small scale data modelling exercise by applying a recently proposed data model (spatially extended entity relationship model, Firns 1994a, 1994b). The data model was published in May 1994, and in fact, it is used in this thesis for the first time since its proposal. Further evaluative work on the spatially extended entity relationship model would lead to potentially useful results as:

- A detailed logical database design and implementation based on the SEER schemas of the case study would prove the usefulness of the conceptual data model.

- Applying different conceptual data model(s) to the same case study design problem would enable the comparison and evaluation of various conceptual schemas.

- Applying the SEER model to other larger design problems would thoroughly test the data model.

6.2 Thesis Conclusion

GIS applications are being used in an increasing number of organisations in New Zealand. These non-traditional applications share the characteristics of traditional data retrieval systems i.e. data is collected and stored before the system can be put to use. Well-designed databases supporting GIS applications are of increasing importance in this new technology. This thesis raises the question: does an appropriate tool (i.e. conceptual data model) exist for the practitioners of GIS database design to help establish such a well-designed database?

This thesis gives a general background to conceptual data modelling. It states that a database is an interrelated collection of data files with the purpose of supporting the data

needs of applications. Before a database is populated, its structure must be designed and built, generally through three phases: the conceptual, the logical and the physical database design phases. The three design phases have three corresponding data models. During the conceptual database design the designer constructs a high level representation of the users' data requirements using a conceptual data model.

After outlining the basic concepts of conceptual data modelling for general purpose databases the thesis introduces all the "specialities" pertaining to GIS databases. The most important specialities/problems are:

- two types of data elements are used to describe spatial features in a GIS database i.e. geographical (spatial) and attribute (descriptive) data elements
- geographical data structures used in representing spatial data in GIS databases and the descriptive data elements are organised according to the logical data model of the selected DBMS,
- the confusion of relating the non-spatial data model taxonomy and the spatial data models
- the applicability of the traditional database phases in a spatial context.

This thesis chooses the objectives of conceptual data modelling as the evaluation criteria for data models encountered throughout the research documented in this thesis, i.e. to model reality and to form the basis for database schema design. A group of published papers is reviewed, selected from proponents of the entity-relationship and of the object-oriented data modelling paradigms and the applications of these data modelling techniques in a spatial context. It compares various extensions to the original entity relationship model, and a comparison of the main data modelling paradigms is included. Data modelling shortcomings encountered in the literature are also summarised. The literature review concludes that not appreciating the conceptual data modelling objectives leads to unsatisfactory conceptual database design.

The selected data model, the spatially extended entity relationship (SEER) model satisfies the data modelling objectives. It is described and applied to produce conceptual schemas for the database design problem of the Manawatu-Wanganui Regional Council. The case study concludes that the SEER model is an appropriate conceptual model in the spatial context, but further evaluative work on the data model would be useful.

The most general conclusion of this thesis is that clear directions showing the guiding principles of conceptual data modelling are essential for practitioners in a spatial context - to avoid the pitfalls of not recognising and separating models of real world objects and their geometric representation.

References

ABITEBOUL, S. and HULL, R. (1987): IFO: A Formal Semantic Database Model, *ACM Transactions on Database Systems,* Vol. **12**, No. 4, pp. 525-565.

ANTENUCCI, J. C., BROWN K., CROSWELL P. L., KEVANY M. J. with ARCKER H. (1991): *Geographic Information Systems :: a guide to the technology,* Van Nostrand Reinhold, New York.

ARANOFF, S. (1989): *Geographic Information Systems: A Management Perspective*, WDL Publication, Ottawa, Canada.

ARMSTRONG, M. P. and DENSHAM, P. J. (1990): Database Organisation Strategies for Spatial Support Systems, *International Journal of Geographical Information Systems,* 1990, Vol. **4**, No 1, pp. 3-20.

ARMSTRONG, M. P., DENSHAM, P. J. and BENNET, D. A. (1989): Object-Oriented Locational Analysis, in: *Proceedings : GIS/LIS'89*, Orlando, Florida, pp 717-726.

BACHMAN, C. W. (1969): Data Structure Diagrams, *Data Base*, Vol. **1**, No. 2 (Summer 1969), pp 4-10.

BATINI, C., CERI, S. and NAVATHE, S. B. (1992): *Conceptual Database Design: An Entity-Relationship Approach*, Redwood City, California: Benjamin Cummings.

BATRA, D. and SRINIVASAN, A. (1992): A Review and Analysis of the Useability of Data Management Environments, *International Journal of Man-Machine Studies,* Vol. **36**, pp 395-417.

BEDARD, Y. and PAQUETTE, F. (1989): Extending Entity/Relationship Formalism for Spatial Information Systems, in: *Proceedings AUTO-CARTO 9,* Baltimore: ASPRS & ACSM, April, pp 819-827.

BEDARD, Y. (1989): Information Engineering for the Development of Spatial Information Systems: A Research Agenda, in: *Proceedings URISA '89*, The 1989 Conference of the Urban and Regional Information Systems Association, Boston, MA, April, Vol. **4**, pp 43-53.

BEKESI, E. and TODD, E. G. (1994): Classification of the Functionality of Geographical Information Systems Software, in: *Proceedings of the Sixth Colloquium of the Spatial Information Research Centre*, University of Otago, Dunedin in association with the New Zealand Computer Society, pp 203-216.

BRACKEN, I. and WEBSTER, C. (1989): Towards a Typology of Geographical Information Systems, *International Journal of Geographical Information Systems,* 1989, Vol. **3**, No. 2, pp. 137-157.

BRODIE, M. L. (1984): On the Development of Data Models, in: BRODIE, M. L., MYLOPOULOS, J. and SCHMIDT, J. W. (eds), *On Conceptual Modelling*, 1984, Springler-Verlag, Heidelberg.

BRYCE, D. and HULL, R (1986): SNAP: A Graphics-based Schema Manager, *IEEE Transaction on Software Engineering*, Vol. **13**, No. 7, pp. 537-549.

BUNDOCK, M. S. (1987): An Integrated DBMS Approach to Geographical Information Systems, in: *Proceedings Auto-Carto 8*, 8th International Symposium on Computer Assisted Cartography, Baltimore, March/April 1987, ASPRS/ACSM, pp 292-301.

CALKINS, H. W. and MARBLE, D. F. (1987): The Transition to Automated Production Cartography: Design of The Master Cartographic Database, *The American Cartographer*, Vol. **12**, No. 2, pp. 105-119.

CASSETTARI, S. (1993): *Introduction to Integrated Geo-Information Management*, Chapman & Hall.

CATTELL, R. G. G. (1991): *Object Data Management; Object-Oriented and Extended Relational Database Systems*, Addison-Wesley Publishing Company.

CATTELL, R. G. G. (ed) (1994): *The Object Database Standard: ODMG-93 (Release 1.1)*, Morgan Kaufmann Publishers, Inc.

CHAMBERS, D. (1989): Overview of GIS Database Design, *ARC News*, Reprint Spring 1989 Issues, Vol. **11**, No. 2.

CHEN, P. (1976): The Entity-Relationship Model - Toward a Unified View of Data, *ACM Transactions on Database Systems*, Vol. **1**, No. 3, pp. 96-153.

CHOI, A. and LUK, W. S. (1992): Using an Object-Oriented Database System to Construct a Spatial Database Kernel for GIS Applications, *Computer Systems Science and Engineering*, Vol. **7**, No. 2, pp 100-121.

CODASYL (1971): *Data Base Task Group Report*, ACM, New York, 1971.

CODD, E. F. (1971): *Normalised Data Base Structure: A Brief Tutorial,* ACM-SIGFIDET 1971, Workshop, San Diego, Calif., Nov, 1971, pp 1-18.

CZEJDO, B., ELMASRI, R. and RUSINKIEWICZ, M. (1990): A Graphical Data Manipulation Language for an Extended Entity-Relationship Model, *Computer*, March, 1990, pp 26 - 35.

DALE, P. F. and McLAUGHLIN, J .D. (1988): *Land Information Management*, Oxford University Press, Oxford, UK.

DATE, C. J. (1991): *An Introduction to Database Systems*, Vol **1**, Fifth Edition, Addison-Wesley Reading, Massachusetts.

DAVIS, C. G., JAJODIA, S., NG, P. A. and YEH, R. T. (eds) (1983): Entity-Relationship Approach to Software Engineering, Elsevier Science Publishers B.V. (North-Holland).

DE MAN, E. W. H. (1988): Establishing a Geographical Information System in Relation to Its Use a Process of Strategic Choices, *International Journal of Geographical Information Systems,* 1988, Vol. **2**, No. 3, pp. 245-261.

EGENHOFER, M. J and FRANK, A. U. (1989): Object-Oriented Modelling in GIS: Inheritance and Propagation, in: *Proceedings AUTO-CARTO 9*, Baltimore : ASPRS & ACSM, April, pp 588-598.

EGENHOFER, M. J and FRANK, A. U. (1992): Object-Oriented Modelling for GIS, *URISA Journal*, Vol. **4**, No.2, Fall, pp 3-19.

EGENHOFER, M. J and FRANZOSA, R. D. (1991): Point-set Topological Spatial Relations, *International Journal of Geographical Information Systems*, 1991, Vol. **5**, No. 2, pp. 161-174.

ELMASRI, R. and NAVATHE, S. B. (1989): *Fundamentals of Database Systems*, Benjamin/Cunnings, Redwood City, California.

ELMASRI, R., WEELDREYER, J. and HEVNER A. (1985): The Category Concept: An Extension to the Entity-Relationship Model, *Data & Knowledge Engineering*, Vol. **1**, 75-116.

ESRI (1990): *PC Understanding GIS: The ARC/INFO Method*, Environmental Systems Research Institute, Inc.

FERNANDEZ, R. N. and RUSINKIEWICZ, M.(1993): A Conceptual Design of a Soil Database for a Geographical Information System, *International Journal of Geographical Information Systems*, 1993, Vol. 7, No. 6, pp. 525-539.

FEUCHTWANGER, M. and POIKER, T. K. (1987): Infological and Datalogical Spatial Models, in: Blais, J. A. R. (ed), *Lecture Notes in Digital Mapping and Land Information*, Department of Surveying Engineering, University of Calgary.

FINKELSTEIN, C. (1989): *An Introduction to Information Engineering: from Strategic Planning to Information Systems*, Addision-Wesley Publishing Company.

FIRNS, P. G. (1990a): Determining a Useful Balance Between Understandability and Rigour in Data Modelling, *New Zealand Journal of Computing*, Vol. **2**, No. 1, pp 13-21.

FIRNS, P. G. (1990b): Entity-Relationship Modelling in GIS Design: An Efficacious Approach or an Exercise in Futility?, in: *Proceedings: AURISA 90, the 18th Annual Conference of the Australasian Urban and Regional Information Systems Association*, Canberra, November 1990, Canberra, Australia: Australasian Urban and Regional Information Systems Association (1990), pp 429-440.

FIRNS, P. G. (1992): Data Modelling for Spatial Information Systems: The Notion of a Spatial Entity, *New Zealand Journal of Computing*, Vol. **4**, No. 1, pp 17-28.

FIRNS, P. G. (1993): The Derivation of Thematic Map Layers from Entity-Relationship Data Models, *The Information Science Discussion Paper Series*, Number 93 / 3, University of Otago, Dunedin, New Zealand.

FIRNS, P. G. (1994a): Extended Entity Relationship Modelling Formalism for the Design Of Spatially Referenced Databases, in: *Proceedings of the Sixth Colloquium of the Spatial Information Research Centre*, University of Otago, Dunedin in association with the New Zealand Computer Society, pp 289-297.

FIRNS, P. G. (1994b): *An Extended Entity Relationship Model Applicable to the Design Of Spatially Referenced Databases*, PhD Dissertation, University of Otago, Dunedin, New Zealand.

FRANK, A. and EGENHOFER, M (1988): *Object-Oriented Database Technology for GIS: Seminar Workbook*, NCGIA.

FRASER, K. G. and TODD, E. G. (1994): The Use of Geographical Information Systems in New Zealand Local Authorities - A Survey Comparing the Situation in July 1993 with February 1992, in: *Proceedings of the Sixth Colloquium of the Spatial Information Research Centre*, University of Otago, Dunedin in association with the New Zealand Computer Society, pp 147-160.

GOH, P. (1988): Spatial Database Updates - A Formal Treatment Using Entity-Relationship Diagrams, *The Australian Surveyor*, Vol. **34**, No. 3, pp. 307-313.

GOODCHILD, M., HAINING, R., WISE, S. and 12 others (1992): Integrating GIS and Spatial Data Analysis: Problems and Possibilities, *International Journal of Geographical Information Systems*, 1992, Vol. **6**, No. 5, pp. 407-423.

GUNTER, O. and LAMBERTS, J. (1994): Object-Oriented Techniques for the Management of Geographic and Environmental Data, *The Computer Journal*, Vol. **37**, No. 1, pp 16-25.

HAINAUT, J-L (1991): Entity-Relationship Models: Formal Specification and Comparison, in: *Kangassalo*, 1991.

HAMMER, M. and McLEOD, D (1981): Database Description with SDM: A Semantic Database Model, *ACM Transactions on Database Systems*, Vol. **6**, No. 2, pp 351-386.

HEALEY, R. G. (1991): Database Management Systems, in: *MAGUIRE et al.*, 1991, Vol. **1**, pp 251-267.

HULL, R. and KING, R. (1987): Semantic Database Modelling: Survey, Applications, and Research Issues, *ACM Computing Surveys*, Vol. **19**, No. 3, pp. 201 - 260.

JACKSON, M. S. (1991) : Tutorial on Object-Oriented Databases, *Information and Software Technology*, Vol. **33**, No. 1, pp 4-12.

KANGASSALO, H. (ed) (1991): *Entity-Relationship: The Core of Conceptual Modelling*, Elsevier Science Publishers B. V.(North-Holland).

KEMP, Z. (1990): An Object-Oriented Model for Spatial Data, in: *Proceedings 4th International Symposium on Spatial Data Handling*, Zurich, International Geographical Union Commission on Geographical Information Systems, 659-668.

KENNEDY, G. J. (1993): *A Systematic Approach to the Specification and Evaluation of an Information Systems Development Methodology*, PhD Dissertation, University of Otago, Dunedin, New Zealand.

KENT, W. (1979): Limitations of Record-Based Information Models, *ACM Transactions on Database Systems*, Vol. **4**, No. 1, pp 107-131.

KIM, W. and LOCHOVSKY, F.H. (eds) (1989): *Object-Oriented Concepts, Databases, and Applications*, ACM Press, Addison-Wesley, Reading, Massachusetts.

KIM, W. (1990): *Introduction to Object-Oriented Databases,* Massachusetts Institute of Technology.

KING, R. (1989): My Cat is Object-Oriented, in: *Kim and Lochovsky*, 1989, pp 23-30.

LAURINI, R. (1991): Introduction to E.R. Modelling For Urban Data Management, in: *Kangassalo*, 1991, pp 475-485.

LAURINI, R. and THOMPSON, D. (1992): *Fundamentals of Spatial Information Systems*, Academic Press Limited.

LENZERINI, M. and SANTUCCINI, G. (1983): Cardinality Constraints in the Entity-Relationship Model, in: *Davis et al.*

LIPECK, U. W. and NEUMANN, K. (1987): Modelling and Manipulating Objects in Geoscientific Databases, in: SPACCAPIETRA, S. (ed), *Entity-Relationship Approach*, Elsevier Science Publishers B.V.(North-Holland).

MAGUIRE, D. J. (1991): An Overview and Definition of GIS, in: *MAGUIRE et al.*, 1991, Vol. **1**, pp 9-20.

MAGUIRE, D. J. and DANGERMOND, J. (1991): The Functionality of GIS, in: *MAGUIRE et al.*, 1991, Vol. **1**, pp 319-335.

MAGUIRE, D. J., GOODCHILD, M. F. and RHIND, D. W. (eds) (1991): *Geographical Information Systems: Principles and Applications (Volume 1 and 2)*, Logman, London.

MCFADDEN, F. R. and HOFFER, J. A. (1988): *Database Management,* Benjamin/Cummings(US).

MILNE, P., MILTON, S. and SMITH, J. L. (1993): Geographical Object-Oriented Databases - A Case Study, *International Journal of Geographical Information Systems*, 1993, Vol. **7**, No. 1, pp. 39-55.

NYERGES, T. L. (1989): Schema Integration Analysis for the Development of GIS Databases, *International Journal of Geographical Information Systems,* 1989, Vol. **3**, No. 2, pp. 153-183.

PERNUL, G. and TJOA, A. (eds) (1992): *Entity-Relationship Approach - ER '92*, Sprienger - Verlag Berlin , Heidelberg.

PEUQUET, D. J. (1984): A Conceptual Framework and Comparison of Spatial Data Models, *Cartographica*, Vol. **21**, pp 66-113.

RABITTI, F., BERTINO, F., KIM, W. and WOELK, D. (1991): A Model of Authorization for Next-Generation Database Systems, *ACM Transactions on Database Systems*, Vol. **16**, No. 1, pp. 90-91.

ROBERTS, S. A. and GAHEGAN, M. N. (1993): An Object-Oriented Geographic Information System Shell, *Information and Software Technology*, Vol. **35**, No. 10, pp 561-572.

SCIORE, E. (1991): Annotations to Support Multiple Kinds of Versioning, *ACM Transactions on Database Systems*, Vol. **16**, No. 3, pp. 419-420.

SENKO, M. E., ALTMAN, E. B., ASTRAHAM, M. M. and FEHDER, P. L. (1973): Data Structures and Accessing in Data-Base Systems, *IBM Syst. J.*, Vol. **12**, No. 1, pp 30-93.

SMITH, J. and SMITH, D. (1977a): Database Abstractions: Aggregation, *Communications of the ACM*, Vol. **20**, No. 6, pp 405-413.

SMITH, J. and SMITH, D. (1977b): Database Abstractions: Aggregation and Generalization, *ACM Transactions on Database Systems*, Vol. **2**, No. 2, pp 105-133.

SMITH, T., MENON, S. , STAR, J. L. and ESTES, J. E. (1987a): Requirements and Principles for Implementation and Construction of Large-Scale Geographic Information Systems, *International Journal of Geographical Information Systems*, 1987, Vol. **1**, No. 1, pp. 13-31.

SMITH, T., PEUQUET, D., MENON, S. and AGARWAL, P. (1987b): KBGIS-II A knowledge-based Geographical Information System, *International Journal of Geographical Information Systems,* 1987, Vol. **1**, No. 2, pp. 149-172.

SPACCAPIETRA, S. (ed) (1987): *Entity-Relationship Approach*, Amsterdam: Elsevier Science Publication.

SPENCER, R., TERorey, T. and HEVIA, E. (1991): ER Standards Proposal, in: *Kangassalo*, 1991.

STAR, J. and ESTES, J. (1990): *Geographical Information Systems : An Introduction,* Prentice Hall.

STEVENS, S. S. (1946): On the Theory of Scales of Measurement, *Science*, Vol. **103**, pp 667-80.

TAGG, R. and MABON, J. (eds) (1992): *Object Management: Data Management 91*, Ashgate Publishing Limited., England, p 5.

TEOREY, T. J., YANG, D. and FRY J. P. (1986): A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model, *ACM Computing Surveys*, Vol. **18**, No. 2, June, 1986.

TOMLINSON, R. F. (1991): The Value and Problems of GIS in Government, *ARC News Fall*, ESRI.

TSICHRITZIS, D. C. and, LOCHOVSKY F. H. (1982): *Data Models*, Prentice-Hall.

WALLIN, E. (1990): The Map as Hypertext: on Knowledge Support Systems for the Territorial Concern, in: *Proceedings on the First European Geographical Information Systems Conference*, Amsterdam, Netherlands, April 10-13, Amsterdam, Netherlands, pp 1125-1134.

WANG, F. and NEWKIRK, R. (1988): An Entity-Relationship Model for Geographical Information System Development, in: *Proceedings : ACSM-ASPRS Annual Convention*, St Louis, March, Vol. **5**, pp 162-172.

WORBOYS, M. F. (1992): A Generic Model for Planar Geographical Objects, *International Journal of Geographical Information Systems*, 1992, Vol. **6**, No. 5, pp. 353-372.

WORBOYS, M. F. (1994): A Unified Model for Spatial and Temporal Information, *The Computer Journal*, Vol. **37**, No. 1, pp 26-34.

WORBOYS, M. F., HEARNSHAW, H. M. and MAGUIRE, D. J. (1990): Object-Oriented Data Modelling for Spatial Databases, *International Journal of Geographical Information Systems*, 1990, Vol. **4**, No. 4, pp. 369-383.

WORBOYS, M. F., MASON K. T. and DAWSON, B. R. P. (1993): The Object-Based Paradigm for a Geographical Database System: Modelling, Design and Implementation Issues, in: MATHER, P. M. (ed) *Geographical Information Handling - Research and Applications*, John Wiley 7 Sons Ltd., England.

YIN, R. K. (1989): *Case Study Research - Design and Methods*, Revised Edition, Applied Social Research Methods Series, Vol. **5**, Sage Publications, Newbury Park.

ZINATELLI, N. and CAVAYE, A. (1994): Case Study Research in Information Systems: Strategies for Success and Pitfalls to Avoid, *New Zealand Journal of Computing*, Vol. **5**, No. 1, pp 17-22.

Appendices

Appendix A

Preparation for the Data Requirements Collection at Manawatu-Wanganui Regional Council (MWRC)

A.1 Objectives

1. Review the existing underground water database (information systems).
2. Investigate the advantages and disadvantages of the existing physical systems.
3. Identify the present and currently foreseeable data needs of the applications used by the groundwater section.
4. Formulate a conceptual schema for the groundwater database, using the SEER (Spatially Extended Entity Relationship) conceptual data model.

A.2 Venue

The data requirements collection will be taking place at the Regional House (MWRC), Palmerston North.

A.3 Time

The data requirement collection must be done during the period 20 - 29 June, 1994 according to the permission of Mr J. P. McDonald, Director of Corporate Services.

A.4 Short Description of the Planned Activities

I am going to start the data requirements collection by interviewing the scientist responsible for the groundwater section. This introductory interview will target the existing database. (Key points: what are the general responsibilities of the groundwater section? What should be reported regularly to the management? What kind of information is given to the other sections and to the public?)

After the interview I plan to observe the operation of the groundwater section and to understand its functions and its place within the organisation.

During the second week, I intend to discuss the further data needs of the database with the user.
In addition to the interviews, I expect to gain access to data files, computer facilities, groundwater section reports, resource consent application forms etc.

A.5 Background Material

Strategic Information Systems Plan for the MWRC,
Report Number 94/EXT/121, March 1994.

MWRC Data Model,
Scientific Software and Systems Ltd, April 1994.

TECHBASE Reference Manual, Version 2.11,
MINEsoft, 1991, Lakewood, Colorado 80228 USA.

A.6 Questionnaire

1. How is the groundwater section & its personnel placed within the organisation?

2. What are the general responsibilities & objectives of the groundwater section?

3. What should be reported regularly to the management?

4. What kind of information is given to the other sections?

5. What kind of information is usually given to the public usually?

6. What is the nature of the data handled by the section?

7. What is the history and present state of the groundwater database?

8. How is data stored in the present system?

9. Does the present system adequately serve the section information needs?

10. In your opinion, how could information technology help to improve the effectiveness of the sections work (i.e. new hardware/software, presently not available data)?

11. In your opinion, what information, not currently supported, will you have to provide in the future (or you would like to provide / you are expected to provide in the present)?

Appendix B

Manawatu-Wanganui Regional Council- Extract from the Strategic Information Systems Plan for the MWRC, Report Number 94/EXT/121, March 1994. -

B.1 Introduction

The Manawatu-Wanganui Regional Council came into being on 1 November 1989. It is an amalgamation of 40 former authorities from within the Manawatu-Wanganui Region. These included Catchment and Regional Water Boards, United Councils, Noxious Plants Authorities, Pest Destruction Boards and Drainage Boards. In addition, it has taken on devolved functions from Central Government. Examples include land transport and under the Resource Management Act 1991 activities such as natural hazard mitigation and hazardous substance control. Fifty-four statues, including much of the Soil Conservation and Rivers Control Act 1941, and all of the Water and Soil Conservation Act 1967 and Town and Country Planning Act 1977 have been superseded by the Resource Management Act 1991.

B.2 The Beneficiaries

The Manawatu-Wanganui Region comprises the area (22179) delineated on S.O. Plan 36010 deposited with the Chief Surveyor of the Wellington Land District. The constituent authorities of the Region comprise:

	Population
Tararua	**19475**
Horowhenua	**29476**
Palmerston North	**70318**
Manawatu	**27182**
Rangitikei	**16649**
Wanganui	**45082**
Ruapehu	**18104**
Parts of Waitomo, Stratford, and Taupo Districts are within the Manawatu-Wanganui Region.	**330**
Region	**226616**

B.3 Representation

The Region is currently divided into eight constituencies. The Council consists of 11 members elected by the electors of the eight constituencies.

B.4 Purpose and Functions

B.4.1 Overall Direction

The Manawatu-Wanganui Regional Council has recognised the need to have a clear focus for its various activities, resulting in a Mission and Goals statement for the Council.

MISSION
TO ENSURE THE NATURAL AND PHYSICAL RECOURCES OF THE MANAWATU-WANGANUI REGION ARE SUSTAINABLY MANAGED FOR THE BENEFIT OF PRESENT AND FUTURE GENERATIONS.

Table B.1 Mission Statement for the MWRC (Strategic Information Systems Plan for the MWRC, Report Number 94/EXT/121, March 1994)

GOALS

- Promote the sustainable management of natural and physical resources in a way, or at a rate, which enables the Region's people to provide for their social, economic, and cultural well-being, and for their health and safety.
- Ensure that the Council takes account of the principles of the Treaty of Waitangi (Tiriti o Waitangi).
- Promote the sustainable use, development, or protection of land.
- Promote the sustainable use, development, or protection of water.
- Avoid, reduce, or remedy the adverse effects of activities upon the environment.
- Ensure the risks to people and communities from natural hazards are minimised.
- Safeguard the life-supporting capacity of air, water, soil and ecosystems.
- Sustain the potential of natural and physical resources to meet the reasonably foreseeable needs of future generations.
- Determine the effects of activities to ensure that natural and physical resources are being managed sustainably.
- Promote the effective management or eradication of agricultural pets and noxious plants.
- Promote a safe, efficient, and cost-effective land transport system.

Table B.2 Goal Statement for the MWRC (Strategic Information Systems Plan for the MWRC, Report Number 94/EXT/121, March 1994)

These goals form the basis of specific objectives for the year in question through the Council's Annual Plan and business plans.

B.4.2 Operating Principles

In working towards the achievement of its mission and goals the Council guided by two sets of principles. Firstly, it must adhere to the statutory mandates set down in legislation. Secondly, there are principles its follows in its corporate conduct.

B.4.2.1 Statutory Mandates - functions, duties, powers and activities under:

- Resource Management Act 1991;
- Soil Conservation and River Control Act 1941;
- Civil Defence Act 1983;
- Agricultural Pest Destruction Act 1967;
- Noxious Plants Act 1978;
- except as otherwise provided Local Government Act 1974;
- except as otherwise provided Transit New Zealand Act 1989;
- except as otherwise provided Transport Services Licensing Act 1989;
- except as otherwise provided any public Act relating to MWRC;
- except as otherwise provided any local Act relating to MWRC;
- Treaty of Waitangi;
- Equal Employment Opportunity.

B.4.2.2 Corporate Conduct - indicates how the Council will carry out its duties (functions, duties, powers and activities).

The Council aims to fulfil its statutory obligations in a manner which takes into account:

- the three inseparable parts to policy development: individuals, society, and the ecosystems of which humans are a part;
- the need to ensure the sustainable management of natural and physical resources;
- the need to provide an environment for physical, social and economic development of the Region;
- the need to recognise the Region's rich and varied history and unique topography and geology;
- the need to fund the Council's activities at a level sustainable by, and not be an unreasonable burden on, the ratepayer.

B.4.3 Functional Activities

The Council's activities are grouped into output classes as follows:

- Representation
- Resource Policy
- Land Transport
- Resource Consents
- **Resource Monitoring and Investigations**
- River and Drainage Engineering
- River and Drainage Schemes
- Soil Conservation
- Noxious Plants Control
- Agricultural Pest Management

B.5 Resource Monitoring and Investigations

B.5.1 Description

Resource Monitoring and Investigations investigates, monitors, analyses and reports on

natural and physical resource and consent compliance so that information/advice for resource monitoring purposes can be provided and performance in the management of natural resources can be monitored.

B.5.2 Overall Objective

Monitor, analyse and report on consent compliance, pollution incidents and environmental quality in the Region to provide information and set standards for Resource Management purposes.

B.5.3 Outputs

1. Permit Compliance
Undertake compliance monitoring of discharges to water, land , air and monitoring of land use permits for gravel extraction in accordance with the specified annual programme. Investigate and report on activities that impact on environment in the Region to provide advice for initiating and processing consents.

2. Environmental Monitoring
Undertake baseline monitoring of surface water quality, groundwater levels and quality, monitoring of hydrometric characteristics of catchments to provide information for resource management, flood prediction, river control, emergency response and scheme reviews. Carry out appropriate investigations on contaminated sites.

3. Impact Monitoring
Impact monitoring includes monitoring the effects of activities on water quality, groundwater levels.

4. Pollution Incidents and Hazardous Substances
Investigate, monitor and resolve pollution incidents. Investigate the establishment of a facility to store hazardous waste for a future collection of redundant agricultural chemicals in the Region.

5. Laboratory
Maintain a water and waste water testing laboratory to provide a high quality sample analysis and data base for Council monitoring programmes.

6. Planning
Initiate work on strategies required by Regional Land Resource Management Plan.7. Advisory
Provide technical input and advice into permits, land use consent applications and By-Law approvals. Investigate and report on various flood control proposals and provide civil engineering advice to territorial authorities and the public.

8. Standard Setting and Monitoring
Establish performance standards for the Council's land resource management activities.

Appendix C

C.1 Organisation Chart

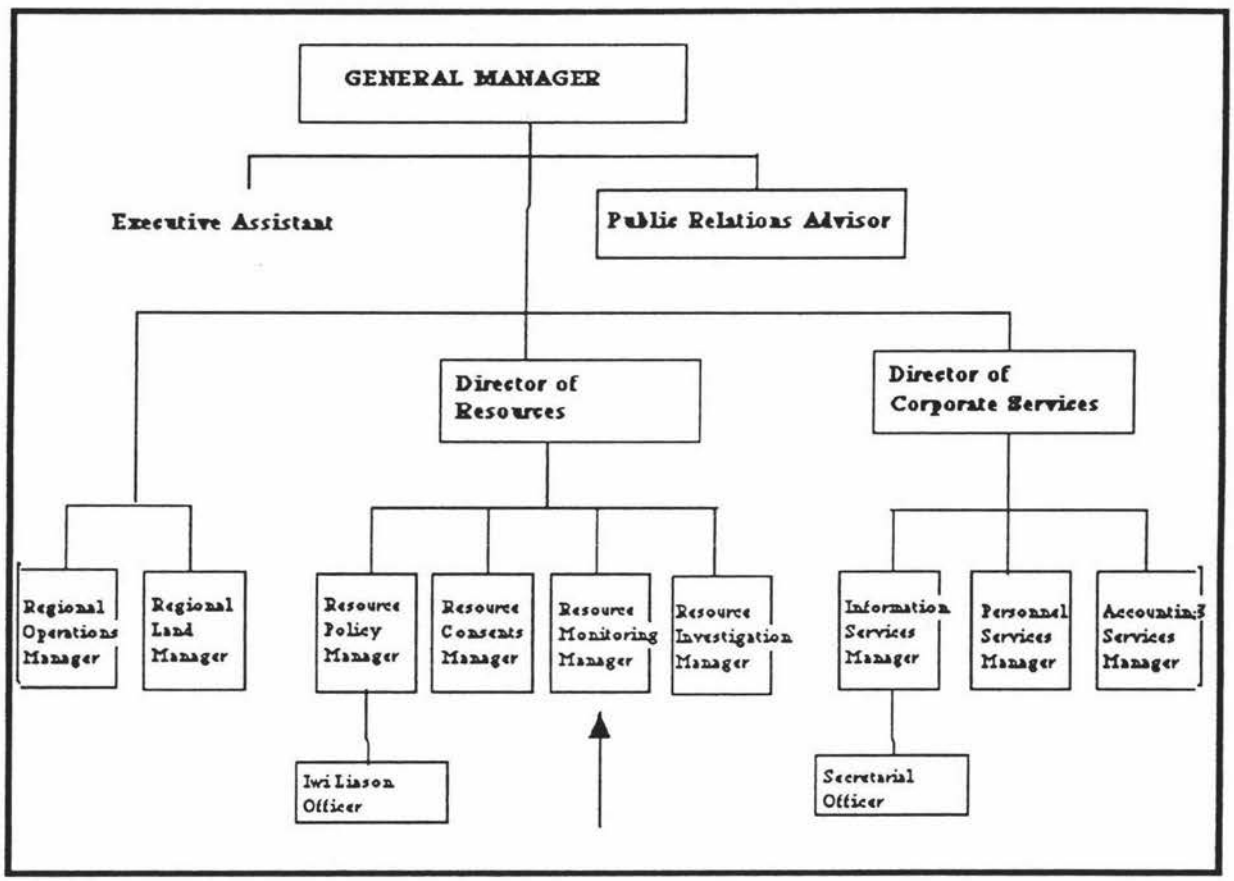


Figure C.1 MWRC Organisation Chart (Strategic Information Systems Plan for the MWRC, Report Number 94/EXT/121, March 1994)

C.2 Resource Monitoring Department

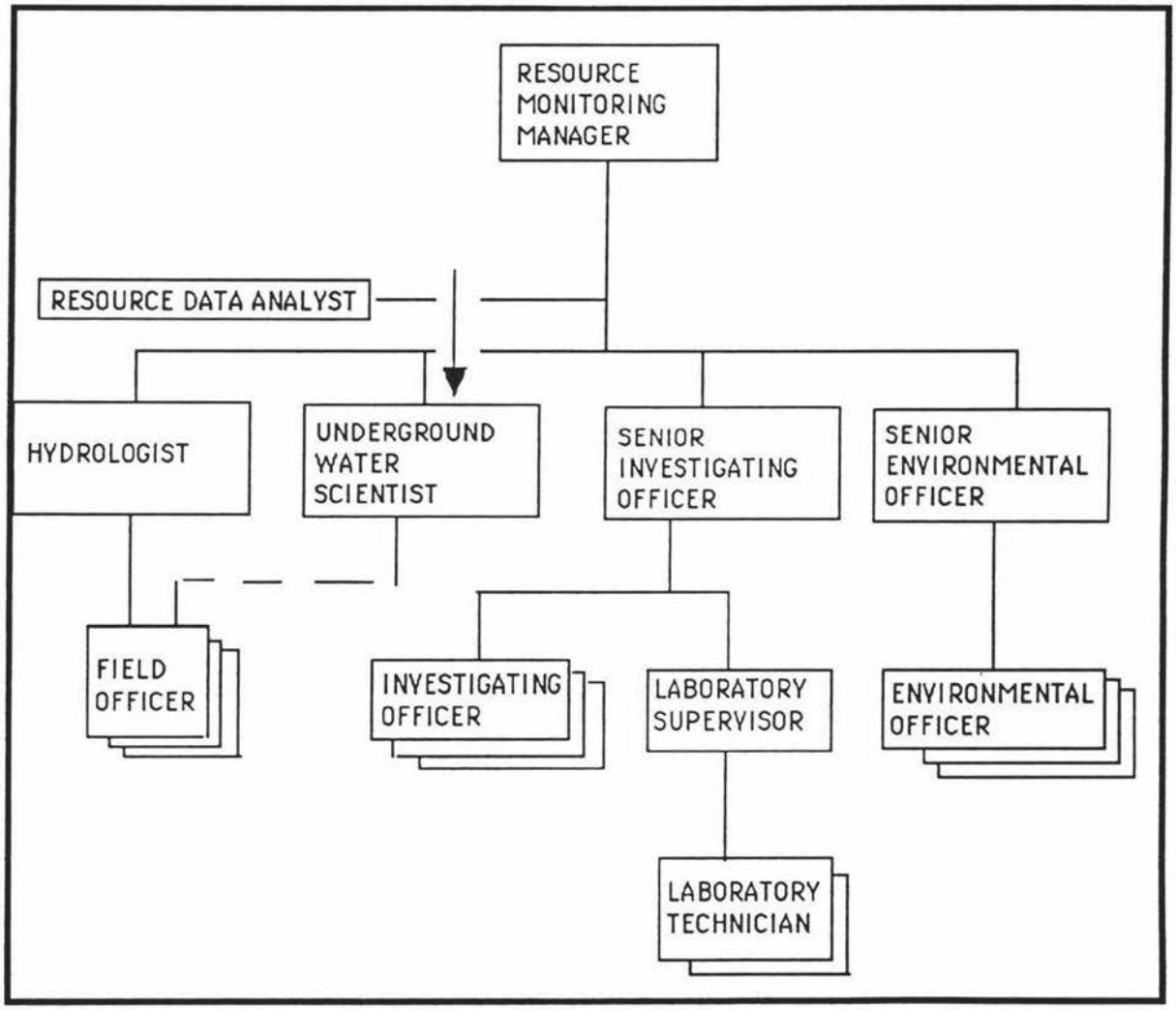


Figure C.2 MWRC Resource Monitoring Department (Strategic Information Systems Plan for the MWRC, Report Number 94/EXT/121, March 1994)

Appendix D

Report on the Data Requirements Collection

D.1 Introduction

The Manawatu Wanganui Regional Council (MWRC) is responsible for the management of underground water[1] resources in its region.

The aim of this report is to document the results of the data requirements collection exercise carried out at the Manawatu-Wanganui Regional Council, in June 1994. The main objective of the exercise was to establish the basic requirements for the Groundwater Section database. The database should be able to support all present and currently foreseeable data needs of the section and the requirement analysis produced in this report should enable the formulation of a conceptual schema for such a database.

D.2 Introducing - Groundwater Management

Groundwater management provides for the sustained usage of groundwater resources without triggering undesirable environmental effects. Harmful effects could range from aquifer contamination to irreversible groundwater level decline. This section of the report describes the major components and the main operational duties of groundwater management.

D.2.1 The three major components

The three major components of groundwater management are: allocation, monitoring and modelling.

D.2.1.1 Groundwater allocation

Water permits (consents) are required only for noticeable groundwater drawings from bores. Industrial users, small water supply schemes for more than five households or anyone who needs more than 15,000 litres water per day from underground aquifers has to apply for a water permit. A water permit authorises the drawing of a maximum amount of groundwater, at a given well, for a limited time. Once that period has elapsed, the permit needs to renewed. The renewed permit may differ from the previous one. Currently there are about 550 permits issued for taking of groundwater in the region. The vast majority of the estimated 10,000 bores in the region are used for stock, domestic and dairy farm purposes without the necessity of a water permit. The Council has the authority to impose restrictions on consents and/or water usage in certain areas by means of laying down Regional Plans or Regional Rules.

---

[1]The words groundwater and underground water are used interchangeably.

D.2.1.2 Groundwater monitoring

Groundwater monitoring is the regular measurement of underground water level and groundwater quality. Monitoring serves as a feedback for allocation. If more groundwater is drawn from under the ground than is replenished naturally, groundwater levels decline.

D.2.1.3 Groundwater modelling

"A model may be defined as a simplified version of the real (here groundwater) system that approximately simulates the excitation-response relations of the latter", definition is given by a textbook[2] well-known in its field. Groundwater models are used to predict the outcome of a problematic situation in the groundwater management domain.

At the MWRC, mathematical and numerical models are applied. How the modelling is done? The first step in the procedure of modelling is the construction of a conceptual model[3] of the problem. The second step is the derivation of the mathematical model from the conceptual model. A very important part of this activity is known as the identification problem, that is finding the coefficients and parameters of the model. Once the mathematical model has been formulated in terms of relevant state variables[4], it has to be solved, either by analytical or by numerical methods. In general numerical methods utilise computer software of some kind.

It is interesting to note that a numerical model is often validated by comparing its predictions with those obtained analytically from the mathematical model. Because a model is only an approximation of the reality, the predicted values of the model are never expected to be identical to the set of values produced by the real groundwater system. Instead, scientists search for the "best fit" between them. Another important feature of groundwater modelling is the uncertainty. The modeller is usually uncertain about whether the selected conceptual model (i.e. his set of assumptions) indeed represents what happens in the real life system.

Groundwater modelling answers pressing questions for the region, such as, what happens to groundwater levels if more groundwater is extracted from an aquifer or what happens to the quality or chemical constituents of groundwater if a new waste disposal site becomes operational. Another important aspect of modelling is to forecast groundwater levels in the case of extreme drought.

D.2.2 Operational duties

Water permits are accredited and Regional Rules are administered by the Consent Department. Remaining operational duties for groundwater management are executed by the Resource Monitoring Department, mostly by the Underground Water Section[5] of the department. The section has one employee, the underground water scientist. His job description defines the responsibilities of the section:

---

[2]Bear, J. and Verruijt, A. (1990): *Modeling groundwater flow and pollution*, D. Reidel Publishing Company, Dordrect, Holland, page 12.

[3]In this context a conceptual model is a set of assumptions expressed in words, a tool to convey the modeller 's understanding of reality. This term is not analogous to the term conceptual data model.

[4]For example: groundwater level can be a relevant state variable in a model.

[5]Some of the groundwater field data are collected by other Resource Monitoring Department staff members.

"The section is responsible for the design, undertaking and overseeing the monitoring, investigation, analysis and reporting of underground water resources in the Manawatu-Wanganui Region. The underground water scientist is also responsible for the design and maintenance of the groundwater archive."

Four points are of interest for the purpose of this report:

• Groundwater monitoring: design, oversee, undertake and report on the Council's groundwater level, groundwater quality programmes. Process all manually collected groundwater level and quality data.
• Groundwater archive: design and maintain a computerised bore archive which includes general bore and groundwater data. Analyse and report on groundwater data to provide information to management, Consents Department and to the general public.

• Permit investigation: oversee and undertake investigations and make recommendations on permit applications made to take underground water.

• Compliance monitoring: oversee and undertake monitoring of major groundwater permits and resolve pollution complaints.

Some explanation on the acquisition of groundwater modelling data in order, because these activities are specific to the underground water management. Some of the data needed for modelling is collected by conducting pump tests. In principal, when a bore is drilled, or when it becomes necessary, a bore is tested. First, the natural or "not pumped" groundwater level is measured. After the static water level has been found out the discharge is measured as a function of time. From these measurements the drawdown, the difference between the pumped and unpumped water level, is calculated. Using the discharge and drawdown data the hydraulic conductivity and the coefficient of storage (groundwater retention in the soil) can be determined. These are vital to groundwater modelling.

Groundwater management is difficult without adequate groundwater data management: allocation, monitoring and modelling all demand reliable and scientifically defensible data. Advice is given to ratepayers about groundwater availability daily. Advice is as good as the data it is based on. Groundwater is vulnerable to surface or sub-surface contamination, the environmental impact assessment of pollution is as correct as the data these appraisals are built on. The next sections of the report examine the source of presently available groundwater data and outline the form in which the groundwater data have been stored since the 1960's.

D.3 The source of groundwater data

Until 1990, groundwater data was collected by the Manawatu Catchment Board and the Rangitikei-Wanganui Catchment Board. In 1989, the two Boards became Central District Catchment Boards and in 1990, the MWRC became the groundwater manager within the region. The Groundwater Section acquires its data from diverse sources. The source of bore data is one or more of the following:

- borelog, documented by the welldriller,
- inspection cards, produced by MWRC (or catchment board) staff,
- catchment bore maps (Rangitikei-Wanganui Catchments only),
- old photo maps (Manawatu Catchment only).

A borelog lists the following data: driller's name, drilling date, the name of owner of the bore, the location of the bore, depth, diameter, what kind of material encountered during drilling. Bores are not necessarily logged, for about 80 % of bores in the present groundwater database there is no borelog data. Most of the new bores are logged, particularly the ones drilled since 1980. Bore data is stored in the section's database.

Groundwater level data are recorded on various field sheets detailing the date and time of observation, the number of bore or name of the owner and the observed water level. Regularly monitored groundwater level data is stored on the MWRC mainframe computer. This data is shared with the Hydrology Section, while the irregularly monitored groundwater level data is stored in the Groundwater Section database.

Groundwater quality data is held in reports detailing the constituents analysed and their concentration. This data is stored exclusively in the Groundwater Section database.

Groundwater permit data is stored in the Consents Department CONMAN database. The Groundwater Section imports data from it at irregular intervals. Some groundwater usage data can also be found on the inspection cards and in old Manawatu Catchment Board files.

Groundwater is recharged by rainfall and surface water flow. The rate of recharge will depend on, among others, the amount and distribution of rainfall, soil characteristics, geology, topography. This data is collected and held either by other departments of the Regional Council or by external agencies (Crown Institutes, Universities, NZ Meteorological Service.)

D.4 Previous groundwater "databases" - a short historical background

Groundwater data have been gathered since the 1960's and the accumulated data have been stored on cards, on maps or on computer files. The storage formats have always been developed in an ad hoc fashion, formal methods have never been applied to their design.

D.4.1 The dark ages

The earliest manual system used cards. For each bore a card was made and groundwater data obtained from the bore were entered onto the card. About 2000 cards were created, indexed by the bore owner's name. A map, about 3 m wide and high, mounted onto the office wall, served as a spatial index. Bores were represented by colour coded pins corresponding to the depth of bores. Small flags, labelled by the bore owner's name, were attached to the pins.

An obvious shortcoming of this system was that, apart from bore depth and owner's name, no other data were presented on the map, therefore no spatial analysis was possible. Bore indexing by owner's name was not really suitable as ownership changed causing problems with maintaining data integrity. In areas of large bore density flags became unreadable, also there was a tendency for pins to be removed for other uses thereby destroying the index. After some time cards become difficult to read and increasingly difficult to retrieve. Groundwater level, quality and permit data were stored in manual files without any cross-referencing to the bore data.

D.4.2 The grey ages

In 1985 a new, but still manual system was instituted. This was based on the NZMS 1:50,000 maps. Each map was divided up into rectangular blocks or grids of 8.25 km x 11 km. Each grid is numbered on the basis of its minimum longitude and latitude. Within each grid every logged bore[6] has been assigned a sequential index number.

Each bore was represented by a circle on the 1:50,000 scale maps. The depth of the bore, in metres, was written above the circle and the well number (unique within the grid) was printed below. The bore depths are colour-coded, colours designating bore depths i.e. 10 - 20 m yellow, 21 - 50 m green, 50 - 70 m red, 71 -107 m blue, 108-150 m purple, 150-... m gold.

Bores were indexed by the unique number discussed above, consisting of the grid number and, within a grid, sequential well numbers. Groundwater level and quality data were indexed by the same bore number. While this system overcome problems such as bore ownership change and cross referencing, spatial search, except of grid search, was still impossible or very labour and time consuming. Permits were stored in a manual filing system, no effort was made to cross reference to permits. Bores without logs (estimated 80 % of all bores) were not documented at all.

D.4.3 Partly computerised system

While the card and map systems made some type of data retrievals easy, it failed to integrate most of the groundwater data. Storage and retrieval of increasing variety and increasing number of records was cumbersome.

The first set of data computerised was general bore, location, groundwater level and some water quality data for some 200 bores in the Horowhenua district. The objective of this "database" was purely to aid a geophysical survey. DATATRIEVE software was used to create and manage these data files on the Manawatu Catchment Board's VAX computer.

While data retrieval and reporting became easier, DATATRIEVE did not allow spatial search, neither could it produce maps of any kind. Some statistical analysis was available but without related graphic capabilities.

D.4.4 The high TECHBASE age

Since 1991, the manual and partly computerised system described in 4.2 and 4.3 has gradually been computerised. A "GIS" TECHBASE, was purchased to assist the Groundwater Section. By late 1992 all available bore spatial and non-spatial data were computerised from borelogs. Data on bore inspection cards, from the former Rangitikei-Wanganui Catchment Board were numbered, processed and computerised. In 1993 bore Manawatu Catchment photo maps were digitised, processed and loaded onto TECHBASE. All water quality data, from the previous Manawatu Catchment Board area were entered. Permit data, from CONMAN, could be imported to TECHBASE. Geophysical data was transferred from DATATRIEVE. Groundwater chemistry data, from the previous Rangitikei-Wanganui Catchment Board area, was

---

[6]For a logged or properly recorded bore the welldriller's log is available: listing details about bore construction, lithology encountered and hydrological test data. More details on borelog are in Section 8.1.

entered into TECHBASE in late 1993 together with tectonic lines, NZMS 260 map boundaries and grids.

D.5 The existing database

D.5.1 TECHBASE

TECHBASE is a relational database management system, extended with graphics and modelling functions to present and analyse spatially referenced data. While it is easy for beginners to learn and use TECHBASE, it lacks several, standard GIS functions. Overlay, edge matching, network analysis, buffer operations and proximity analysis functions can only be performed by using the programming language of the software. Spatial and descriptive (non-spatial) data are stored in the same files. The software implements the vector data model to store spatial data but it handles raster data too.

The internal files used by TECHBASE are called tables. A table contains a series of records, each with a value for a set of related fields. Each table can be of a variety of types and some table types are structured, i.e. the structure of the table is predefined. The main table types are: flat, cell, layer, block, polygon, edge, vertex, join and cell join tables.
A flat table is an unstructured file type, similar to a spreadsheet and stores unprocessed, raw data. A cell table provides storage for raw data or models represented as a two-dimensional array of records. (Spatial data in raster mode.) A layer table is the same as a cell table, the difference is that this type of table holds a stack of two dimensional arrays of data values. Block tables store three dimensional (x, y, z) arrays of data values. Polygon tables store boundary points and descriptive data associated with either 2D or 3D polygon, in vector data model fashion. A three table storage method allows users access to all components of the polygon. When a polygon table is created, an edge table and a vertex table are named (and created if necessary) to hold polygonal edges and vertices. An edge table can be created independently from a polygon table, to store line segments, and as part of its creation, a vertex table is named to store coordinate values. Vertex tables store 2D or 3D point values. A join table enables two tables to be combined using common field of these tables. A cell join table uses table geometry to join cell, block or layer table with a cell table.

Valid TECHBASE field types are integer, real, text and date. Field classes are actual, calculated, measured and automatic field. An actual field is physically stored in the database, calculated fields are calculated from other fields from the same record. A measured field contains actual values which may have text values with numeric equivalents. It combines the advantages of numeric and text fields. For reporting purposes the text values are used, for arithmetic operations the numeric values are used. Automatic fields are automatically created as part of the table definition.

D.5.2 WELLARC tables and fields - a detailed description

The name of the present groundwater database is **WELLARC**. Bore and groundwater quality data, for approximately 7,000 bores are arranged in twelve different tables in WELLARC. Table D.1 summarises groundwater data stored at the groundwater section. For more detail the reader is referred to Appendix E where the database definition is listed.

NAME OF TABLE	TYPE	COMMENT	KEY FIELDS	NUMBER OF RECORDS
LOCATION	FLAT	BORE LOCATION AND HYDROGEOLOGY	GRID WELL_NO	7010
QUALITY	FLAT	GROUNDWATER CHEMISTRY	GRID WELL_NO DATE_ANA	2040
USAGE	FLAT	MWRC WATER USE		640
WOBS	FLAT	GROUNDWATER LEVEL AND FLOW OBSERVATIONS	GRID WELL_NO DATE_OBS	670
LITHO	FLAT	LITHOLOGY	GRID WELL_NO L_FROM	4700
F_MODEL	LAYER	MANAWATU GROUNDWATER MODEL		126000
MAPGRIDS	POLYGON	NZMS 260 MAPS		28
FAULT	POLYGON	TECTONIC LINES		40
GRIDS	POLYGON	MWRC GRID DATA		100
GRAVITY	FLAT	GEOPHYSICAL DATA		380
RESIST	FLAT	GEOPHYSICAL DATA		1160
TEMAT	FLAT	WHAKARONGO PROJECT	GRID WELL_NO	200

Table D.1 MWRC Groundwater data description

LOCATION

The LOCATION table stores general bore construction, water level and pump test data.

The identifier of the table is a concatenated key, **grid and well_no** (refer to section D.4.2).
In some graphic applications, it is easier to use only one key field. For this reason, a calculated field, **holeid**, has been created (**Holeid** = 1000 * **grid** + **well_no**).

Fields **map, east** and **north** are based on the NZMS 260 1:50,000 scale maps published by the Department of Lands and Survey (now Department of Lands and Survey Information). Map is the sheet name, such as S25. **East** and **north** are four digit map references, ideally precise to the nearest 100 metres. The relationship between **east** and **north**, and between **feast** and **fnorth**, the seven digit "full" NZMS map references: **Feast** = (26000 + **east**) * 100 and **Fnorth** = (60000 + **north**) * 100.

Permit is the consent number, imported from the CONMAN database. If there is no consent attached to the bore, **permit** is blank.

Depth is the bore depth (depth to bottom of aquifer) in metres. **Decol** and **dde** calculated fields are to facilitate colour coded bore depth plotting and cross section plotting respectively.

Driller, dia and **date_drill** are the name of the driller (drilling company), bore diameter in inches and the date of bore drilling (completed).

Swl is the static water level either at the time of bore completion or at the first documented bore inspection. **Altit** is the wellhead altitude, as surveyed in terms of msl (mean sea level) and in metres. **Corr** is the vertical distance between the reference point surveyed to msl and the reference point used for static water level (**swl**) measurements. **Corr** is positive if the survey reference point is below the swl reference. Hence **walt**, groundwater altitude in terms of msl is the sum of **swl**, **altit** and **corr.**

Monitor is a text field which is blank if the bore is not monitored regularly for static water level by the MWRC. Otherwise it is set to the name of the groundwater run the bore is part of, for instance Horowhenua, Opiki or Wanganui. **Recstart, recstop** and **reclen** are the static water level recording start and finish dates and the available record length in years.

Surname and **init** are the surname and initials of the bore owner at the time of bore completion or at last inspection. **Region** is either MWRC or blank if the bore is outside the MWRC area, as in the case of large number of bores around Otaki in the former Manawatu Catchment area. **Area** is one of the major groundwater catchments: Wanganui, Whangaehu-Turakina, Rangitikei, Manawatu, Horowhenua and Eastern area. **Areacol** is numeric field corresponding to the area. **Locality** is location within the area, including Longburn, Kai-Iwi or Eketahuna. **Whaka_mod** is set to "y" if the bore is within the Whakarongo groundwater modelling area, otherwise it is blank.

Testtype is the type of pump test performed on the bore:
S:step drawdown, C:constant rate, P:production test, -O delineates that observation (interference) bore(s) were monitored during the test.

Discharge is the pump test rate, in m^3/day; **drawdown** is the corresponding drawdown observed in metres. **Spdis**, specific discharge, is the ration between discharge and drawdown (if drawdown exist). **Stor** is the coefficient of storage, as calculated from pump test data. **Trans** is the aquifer transmissivity in m2/day.

Slot is the screen slot size in 1/1000 inch, if screen is installed; blank otherwise. **Thick** is the screen length in metres.

Aqcode is the lithological type for the aquifer, like gravel, sand, greywacke or limestone. **Gclass** is a ground class number reflecting the degree of consolidation for strata down to 20 metres depth. This field was assigned by Hugh Cowan and Kelvin Berryman, IGNS, for their earthquake hazard study, commissioned by the MWRC.

Blog is a text filed indicating whether the borelog is available (y) or not (n). **Source** is the source of the information borelog, card or photo. Card means an inspection card, photo means the information source for the record is an old photo map (refer to section D.3).

QUALITY

The QUALITY table stores groundwater quality data.

The key fields are: bore key fields (**grid, well_no**) and **date_ana** field. Date_ana is the date of sampling and it is a date field. (Since groundwater can be sampled and analysed several times from the same bore and groundwater quality can change over time.)

Most of the field names used in QUALITY are self-explanatory, normally the field name is set to its chemical notation, e.g. **Ca** for calcium. **Qcount** is a field set to 1 if the water analysis is the latest for a particular bore (for the same **grid** and **well_no**,

date_ana is the latest), 0 otherwise. This field should be used when maps for water quality constituents are prepared in order to have only one value for a bore.

Cond is the groundwater conductivity in mS/m. **Ca, Mg, tothard** (total hardness) are all in mg/l CaCo3 units. **Fe, Mn, Na, K, Cl, F, Al, Si, Cu, As Cr** and **Zn** are all in mg/l units. **NO2** is the nitrite-nitrogen, **NO3** is the nitrate-nitrogen, **NH3** is the ammoniac nitrogen in mg/l Nitrogen units. Note that in these units, the recommended drinking water standard (USA Environmental Protection Agency) is 10 mg/l NO3.

Fcol, totcol, ecocci and **concol** are the faecal coliforms, total coliforms, Enterocci or confirmed coliforms respectively, in 1/100ml units. **Bod5** is the five day biological oxygen demand.

HCO is the total alkalinity, in mgr/l CaCo3; **CO** is the carbonate alkalinity in mgr/l CaCo3. Tds is the total dissolved solids in mgr/l. **Susp** is the suspended solids in mgr/l units.

Several fields are measured fields, that is, non-detectable or less than detection limit values, are entered as "nd". This makes it easier to differentiate between non analysed (blank), zero (0) or non-detectable (nd) values and ensures that statistical operations are performed on the whole sample base not only on detectable sub-populations.

The QUALITY table has a large number of calculated fields. Calculated fields **eca, eso**, etc. are the observed values, ca, so etc., converted into milliequivalent units. Calculated field **pca**, for example, is the ratio between **ca** and the total cations, **catsum** (both **ca** and **catsum** in milliequivalents). **Sar** the sodium adsorption ratio often used in groundwater quality studies as a measure of corrosiveness of the water.

WOBS - Groundwater level and flow observations

In this table the irregularly monitored water level and flow observations are stored in fields **wlobs** and **flow** respectively. **Wlobs** is the observed static water level in metres, the datum (**wlobs** = 0 m) is the wellhead. **Flow** is the measured groundwater extraction rate, in m^3/day.
The key fields are **grid, well_no** and the date of observation, **date_ana**.

USAGE

This table is imported from the MWRC CONMAN database. Table USAGE contains groundwater permit data. **Permit** is the MWRC consent number (key field in CONMAN). **Holder** is the name of the consent holder. **Userate** is the maximum daily groundwater drawing authorised by the permit. **Date_ana** is the date of permit issue.

Meast and **mnorth** are seven digit easting and northing as imported from the CONMAN database. **Eastc** and **northc** are calculated four digit easting and northing comparable to fields **east** and **north** in table **location.** (easting and northing for the same bore are not necessarily the same in CONMAN and WELLARC) **Grid** and **well_no** fields are added to this table but, unlike in the other tables, they are not key fields.

LITHO

The **LITHO** table stores bore lithology data, if available. **L_from** is the depth, in metres, to the top of the strata, **L_to** is the depth to the bottom. **Lith1**, **lith2**, and **sub1** are the primary, secondary and descriptive lithology fields. The codes used in these fields are:

G gravel, conglomerate;
S sand, sandstone;
Z silt, siltstone;
C clay, claystone;
L limestone, shells;
P peat;
O organic ;
V volcanic rock.

For a unit which is described by the welldriller as gravel and silty sand, **lith1** = g, **lith2** = s and **sub1** = z. **Descr** is the text field containing the welldrillers' description of the strata.

Aquifer is set to the name of the aquifer (if known) or blank if the strata does not contain enough water. **Siz** is set to 1 if **aquifer** exist or to 0 otherwise. **Aqseq** is a numeric field which represents unique pen numbers (colours) for each named **aquifer**. **Age** is the age of the strata, in years, if known.

L_patt is the TECHBASE fill-pattern in number; this field facilitates the graphical representation of fields **lith1** and **lith2**. This table has three key fields: **grid, well_no** and **l_from**.

F_MODEL

The F_MODEL table stores data which required by the FLOW groundwater model. This table is a layer table with five layers. Each represent an aquifer in the Manawatu area. Data stored in this table is derived from other tables. This table is being built at present and changes both to the table structure and data records are implemented daily. For this reason this table is not described in detail here.

FAULT

This table stores tectonic lines: faults and anticlines. **Name** is the name of the tectonic line, **tect** is the type (whether fault or anticline).

MAPGRIDS and GRIDS

Table MAPGRIDS stores the polygons surrounding each of the NZMS 260, 1:50 000 scale maps published by DOSLI. The GRIDS table store polygons which enclose grids.

GRAVITY and RESIST

These tables store geophysical data, gravity measurements in **GRAVITY** and electric resistivity measurements in the RESIST table. Both tables store geophysical measurements and the site of those measurements.

TEMAT

This table stores Te Matai Road, Whakarongo project groundwater data and includes some imported (duplicated) data from table LOCATION. Data from this table will be transferred to table WOBS.

After the groundwater management and its scope have been defined, the source, the history and the current storage format of the groundwater data have been explored, the report continues to describe the identified data requirements of the Groundwater Section.

D.6 Data Requirements of the Underground Water Database

Inquires about groundwater are received daily by the MWRC from ratepayers, potential bore owners, drilling companies, environmental groups, consultants and territorial authorities.

D.6.1 Simple inquiries

Simple inquiries are those which obligate little processing of the data in stored in WELLARC. These inquiries normally relate to bores rather than the environmental impacts. The most frequently asked questions are:

- What data are available for a given bore?
- How deep is a bore which belongs to an owner at given location?
- How much water can a bore produce?
- Who owns a bore at given location?
- How deep should a bore be to produce certain amount of water at a given location?
- Where is the nearest bore?
- How many bores are there within an area?
- How much water is permitted to be taken from a certain bore?
- What is a particular bore used for (i.e. stock, domestic, dairy shed, irrigation, industrial or municipal water supply) and who is using the water?
- Is there a bore with a large groundwater extraction permitted?
- Does the MWRC has any restrictions on groundwater allocation within an area?
- What is the groundwater quality like at a given location and/or depth?
- How does groundwater quality change, at a give location, with bore depth?
- How does groundwater quality change in time in a given bore?
- Is groundwater suitable for a particular use?
- Is the water from a bore drinkable?
- If groundwater quality is poor, which constituents do not meet criteria for drinking or irrigation water standards?

D.6.2 Complex inquiries

Complex inquiries require interpretation of raw data include:

- A property is about to be purchased, how much groundwater and what quality is available under the property?
- What is the total use of groundwater in a catchment?[7] What was the same 10 years ago?

---

[7]The region has been divided to groundwater catchments or areas for statistical purposes, refer to D.5.2.

- Is there a water level decline in an area or for an aquifer?
- Where (within a property) and how deep to drill a new bore to satisfy given criteria for obtaining drinking quality water?
- Which is the best aquifer, at a given location for a certain groundwater use ?
- What is the environmental impact of an existing groundwater drawing?
- What is the likely impact of a proposed groundwater extraction?
- Is a water permit likely be issued for a new bore at a location? Are there groundwater allocation problems in an area?
- How many bores will be interfered with by a proposed groundwater take and which are those bores?
- What is the cause of a decrease in groundwater taking in a bore at a given time?
- What bores are likely to be affected by a contamination event at a given location?
- Who owns those bores and what are they used for?

D.6.3 Non-inquire based tasks

D.6.3.1 Reports to management

Routine underground water reports are not submitted to the management. The most frequent request from management is to report on the groundwater resources of a specified area. These reports contains narrative description of underground water resources, the most important points usually are:

- Whether demand for groundwater exceeds the supply?
- Available quantity and quality of groundwater?
- If there is any problem in groundwater allocation?
- The recharge area (the source of groundwater)?
- MWRC monitoring sites and the up to date results of monitoring?
- If groundwater is vulnerable for surface or subsurface contamination?

D.6.3.2 Environmental impact assessments - information given to Consent Department

When an application for a groundwater permit is lodged, the MWRC has to assess the impact of the proposed take on the environment. This assessment includes the estimation of interference between bores. An appraisal typically answers questions such as:

- Where is the applicant's bore and how deep is it?
- Which groundwater area (catchment) is the applicant's bore belong to?
- What is the daily groundwater quantity to be taken ?
- What is the static water level in the bore?
- Where are the nearest MWRC groundwater monitoring sites?
- Who are the neighbours, who own bores, of the applicant?
- What is the cumulative environmental impact (the impact of all uses within a catchment?
- Is there any pump test data (discharge, drawdown, test type) available ?
- What is the likely interference (the impact of the application on surrounding bores) ?
- What conditions are to be imposed on the application, if granted ?

D.6.3.3 Aquifer vulnerability studies - information given to Consent, Policy and Monitoring Department

Surface or subsurface discharges can contaminate groundwater resources (aquifers). In order to assess aquifer vulnerability, among others, the following questions must be answered:

- What type of soil covers the area of interest?
- What is the topography (land relief) like?
- What is the depth to water table (static water level in unconfined aquifers)?
- What is the recharge (rainfall, evapotranspiration, runoff) over the area of interest?
- What is the vadose zone (strata between the soil and the water table) like?
- What is the hydraulic conductivity (transmissivity) of the aquifer
- What is the aquifer media ?
- Which way does groundwater flow ?

Some of the data can be found in the groundwater database (aquifer media, static water level, transmissivity), some within the MWRC (rainfall, runoff) and some data are collected by other organisations. In order to assess aquifer vulnerability, several, otherwise disparate spatial data sets must be related, for example by performing overlay operations.

D.7 Data deficiencies - from the user perspective

In this section data deficiencies, as experienced by the user and relevant to the data requirements identified in section 6, are discussed. Crucial problems with the existing database, as the user summarised, include the lack of precise locational (easting, northing) data, data redundancy and update anomalies between CONMAN and WELLARC databases, the lack of reliable bore ownership data and generally the lack of digitised cadastral, road, soil, land use and hydrography data.

D.7.1 Bore locations and ownership

It is clear from section 6 that groundwater inquiries are always concerned with a bore, site or area. Bore sites are described by the inquirer:

- on the land owned by someone and/or
- with respect to the nearby roads, streams or rivers.

The current procedure, to find a bore described by location or by owner, is cumbersome:

- The bore is found on 1:50,000 scale hardcopy maps,
- map references, i.e. easting and northing are noted,
- search around the map references is performed on the computer and
- record is retrieved (hopefully).

The owner described by **surname** and **init** fields in table **location** is not necessarily the current owner of the bore. In order to find a current owner, the following data are needed:

- land parcel data including boundary polygons in digital format,
- bore easting and northing precise to the nearest metre.

The first step above requires the use of up to date, digital cadastral maps. The best way to acquire data of the second step is by using sub-meter global positioning systems or GPS in the underground water scientist's opinion. In order to describe a bore position, in terms of the nearby roads or rivers, digital road and hydrographical thematic maps are needed.

Digital road, land parcel and hydrography data would enable the user to search on the computer screen by simply overlaying bore data and other maps.

D.7.2 Groundwater permits and groundwater use

There are two major problems associated with groundwater permits and use in the current form of WELLARC: data redundancy between CONMAN and WELLARC and the lack of information on groundwater use and user.

Data redundancy
The MWRC CONMAN database stores details for underground water permits. The key field for the CONMAN database is the consent number, **consent_no**. Several different consent numbers can refer to the same bore as the result of successive renewals of the groundwater permits. At present **consent_no** (the latest for a bore) is duplicated in WELLARC (**permit**) by importing data from CONMAN to WELLARC from time to time. This means that the contents of permit field in WELLARC are not updated automatically when CONMAN data is being updated.

Groundwater usage from a bore
Groundwater usage which does not require permits (refer to D.2.1) is not documented at all in WELLARC. This information is essential for answering the simplest enquire i.e. what for used a bore.

Groundwater users
Although the owner of a property generally owns the bore situated within the property, the user(s) of the bore can be different to the current owner. The owner and/or principal user may decide to supply groundwater to the neighbour(s). Such arrangements, even if legal (water easements), are not documented in CONMAN neither in WELLARC.

D.7.3 The lack of reference data

As described in 6.1 most clients refer to location with respect to the nearest roads, streets, major fence lines, drains, water courses which can be readily seen on maps. The lack of digitised road, cadastral and hydrographic data prohibit efficient, on-screen searches and severely limit the quality of outputs: maps, cross sections and three dimensional pictures.

D.8 Conclusions

This report summarised the results of the data requirements collection carried out at the Manawatu Wanganui Regional Council exclusively for data modelling exercise purposes.

The report introduced the groundwater management and the Groundwater Section's responsibilities, reviewed the sources of the section's data, gave background to the

history and the present stage of its database. The used software, TECHBASE was described. Present data requirements of the section and conjunction with the present data deficiencies were identified.

Appendix E

Database Definitions for the Present Underground Water Database

DATABASE definition for database = wellarc
Title: mwrc well archive
Created: 1992/02/12;13:47 Modified: 1994/06/30;14:47

TABLES:
location is FLAT; with 60 fields, 7015 records, 2055395 bytes
Title: BORE LOCATION AND GENERAL DESCRIPTIVE DATA Created: 1992/02/12;13:47
Modified: 1994/06/24;08:43 Data: 1994/06/30;14:47

Table Keys:
grid well_no

Fields:

location_rec	location_nul	*grid	*well_no
east	north	map	depth
&altit	&swl	&corr	monitor
old	surname	init	+alev
+walt	area	discharge	drawdown
testt	aqtype	stor	transmis
+spdis	slot	thick	CODE
LOCALITY	region	+decol	+HOLEID
+DDE	recstart	RECSTOP	+RECLEN
+INTCOL	GCLASS	blog	AREACOL
+FEAST	+FNORTH	aqcode	WHAKA_MOD
source	permit	WHAQCOL	+SWCODE
+DECODE	AQNUMB	userate	DRILLER
DIA	DATE_DRILL	LOCD	W_COL
W_ROW	+COL	+ROW	+td_cm

quality is FLAT; with 73 fields, 2043 records, 437202 bytes
Title: GROUNDWATER CHEMISTRY DATA
Created: 1992/02/12;13:47 Modified: 1994/06/01;16:25 Data: 1994/06/24;09:29

Table Keys:
dateana grid well_no

Fields:

quality_rec	quality_nul	*dateana	ph
temp	cond	ca	mg
tothard	&fe	&mn	na
k	&co	hco	cl
&so	&f	&co2free	&no2
&no3	&po4	tds	&nh3
&al	&si	&cu	*grid
*well_no	&turb	+date_year	+mgtoca
+natocl	+hardtokna	+natok	ABS
&FCOL	&totcol	&concol	&BOD5
&SUSP	qcount	&boron	&cr
&as	phs	colour	Zn
+ECA	+EMG	+ECAMG	+EK
+ENA	+ENAK	+CATSUM	+PCA
+PMG	+PCAMG	+PNA	+PK

```
   +PNAK          +EHCO          +ECL          +ESO
   +ANSUM          +PCL          +PSO          +PHCO
   +sar          &ECOCCI         TRMEAN         TRDEV
    D18O

temat      is   FLAT; with  22 fields,    201 records,    19095 bytes
   Title: WHAKARONGO PROJECT DATA
 Created: 1992/02/12;13:47  Modified: 1994/06/24;08:43  Data: 1994/02/08;15:51

  Table Keys:
grid          well_no

  Fields:
    temat_rec       temat_nul       *grid          *well_no
    aqcode          thick          se91           ju91
   +sep91          &altit          +jun91          ma92
   +MAR92           east           north          depth
    JUL92          +JULY92          FEB93         +FEBR93
    WL             &swl

usage      is   FLAT; with  12 fields,    642 records,    50076 bytes
   Title: WATER USAGE DATA IMPORTED FROM CONMAN
 Created: 1992/02/12;13:47  Modified: 1994/02/09;07:19  Data: 1993/09/23;14:41

  Table has no Keys

  Fields:
    usage_rec       usage_nul       meast          mnorth
    userate        +easte          +northe         type
    permit          HOLDER          grid           well_no

wobs       is   FLAT; with  7 fields,    671 records,    11424 bytes
   Title: RANDOM WATER LEVEL AND FLOW OBSERVATIONS
 Created: 1992/02/12;13:47  Modified: 1994/02/09;07:23  Data: 1994/06/02;10:57

  Table Keys:
grid          well_no       dateana

  Fields:
    wobs_rec        wobs_nul        WLOBS          *grid
   *well_no         flow           *dateana

mapgrids   is POLYGON; with  15 fields,     28 records,     672 bytes
   Title: NZMS 260 MAP DATA
 Created: 1992/02/12;13:47  Modified: 1993/06/22;14:24  Data: 1993/05/31;16:37

  Table attributes:
   Edge table = mapgrids_e   Dimension = 2
  Table has no Keys

  Fields:
    mapgrids_rec     mapgrids_nul     mapgrids_npt     mapgrids_edg
    mapgrids_xc      mapgrids_yc      mapgrids_are     mapgrids_per
    mapgrids_xmn      mapgrids_ymn      mapgrids_xmx      mapgrids_ymx
    mg_id           mg_col           mg_style

DISCH      is   FLAT; with  9 fields,    1393 records,    62685 bytes
   Title: MWRC DISCHARGES IMPORTED FROM CONMAN
 Created: 1992/02/12;13:47  Modified: 1993/06/22;14:24  Data: 1993/04/20;16:00

  Table has no Keys

  Fields:
```

```
   DISCH_rec      DISCH_nul      meast          mnorth
   DISRATE        where          permit         +easte
  +northe

resist    is  FLAT; with  8 fields,   1164 records,    52425 bytes
   Title: RESISTIVITY SOUNDING DATA
 Created: 1992/02/12;13:47  Modified: 1993/06/22;14:24  Data: 1993/11/09;15:31

  Table has no Keys

  Fields:
   resist_rec      resist_nul      east           north
   sounding        scode           abhalf         ra

grids     is POLYGON; with  15 fields,    97 records,     1380 bytes
   Title: MWRC GRID DATA
 Created: 1992/02/12;13:47  Modified: 1993/06/22;14:24  Data: 1993/09/27;08:55

  Table attributes:
   Edge table = grids_e  Dimension = 2
  Table has no Keys

  Fields:
   grids_rec       grids_nul       grids_npt      grids_edg
   grids_xc        grids_yc        grids_are      grids_per
   grids_xmn       grids_ymn       grids_xmx      grids_ymx
   grid_id         grid_col        grid_sty

mapgrids_v  is  VERTEX; with  4 fields,    864 records,    14688 bytes
   Title: needed by EDGE
 Created: 1993/04/21;10:39  Modified: 1993/04/21;10:39  Data: 1993/04/21;10:39

  Table attributes:
   Minimum X coord:        0  Maximum:        1  Tolerance: *
   Minimum Y coord:        0  Maximum:        1  Tolerance: *

  Fields:
   mapgrids_v_rec    mapgrids_v_nul    mapgrids_v_xc    mapgrids_v_yc

mapgrids_e  is   EDGE; with  9 fields,    758 records,    12090 bytes
   Title: needed by POLYGON
 Created: 1993/04/21;10:39  Modified: 1993/04/21;10:39  Data: 1993/05/31;16:37

  Table attributes:
   Vertex table = mapgrids_v  Dimension = 2

  Fields:
   mapgrids_e_rec    mapgrids_e_nul    mapgrids_e_v1    mapgrids_e_v2
   mapgrids_e_nxt    mapgrids_e_xc1    mapgrids_e_xc2   mapgrids_e_yc1
   mapgrids_e_yc2

grids_v    is  VERTEX; with  4 fields,    138 records,    4692 bytes
   Title: needed by EDGE
   Created: 1993/04/21;10:39  Modified: 1993/04/21;10:39  Data: 1993/09/27;08:55

  Table attributes:
   Minimum X coord:        0  Maximum:        1  Tolerance: *
   Minimum Y coord:        0  Maximum:        1  Tolerance: *

  Fields:
   grids_v_rec       grids_v_nul       grids_v_xc       grids_v_yc

grids_e    is   EDGE; with  9 fields,    372 records,    10374 bytes
```

```
  Title: needed by POLYGON
 Created: 1993/04/21;10:39  Modified: 1993/04/21;10:39  Data: 1993/09/27;08:55

 Table attributes:
   Vertex table = grids_v   Dimension = 2

 Fields:
   grids_e_rec      grids_e_nul      grids_e_v1       grids_e_v2
   grids_e_nxt      grids_e_xc1      grids_e_xc2      grids_e_yc1
   grids_e_yc2

GRAVITY    is   FLAT; with   8 fields,    378 records,    12474 bytes
   Title: GRAVITY DATA
 Created: 1993/05/26;16:51  Modified: 1993/06/22;14:24  Data: 1993/05/27;09:16

 Table has no Keys

 Fields:
   GRAVITY_rec       GRAVITY_nul       meast             mnorth
  +eastc             +northc           RESID             BOUGER

lay_type   is   FLAT; with   4 fields,      5 records,       30 bytes
   Title:
 Created: 1993/07/29;11:25  Modified: 1993/07/29;11:27  Data: 1994/04/05;16:01

 Table has no Keys

 Fields:
   lay_type_rec      lay_type_nul      lt                antrpy

fault_v    is  VERTEX; with   4 fields,    138 records,     2346 bytes
   Title: needed by EDGE
 Created: 1993/09/29;14:51  Modified: 1993/09/29;14:51  Data: 1993/09/29;14:52

 Table attributes:
   Minimum X coord:          0  Maximum:    99999999  Tolerance: 5.0000e+001
   Minimum Y coord:          0  Maximum:    99999999  Tolerance: 5.0000e+001

 Fields:
   fault_v_rec       fault_v_nul       fault_v_xc        fault_v_yc

fault_e    is   EDGE; with   9 fields,     98 records,     1274 bytes
   Title: needed by POLYGON
 Created: 1993/09/29;14:51  Modified: 1993/09/29;14:51  Data: 1993/09/29;14:52

 Table attributes:
   Vertex table = fault_v   Dimension = 2

 Fields:
   fault_e_rec       fault_e_nul       fault_e_v1        fault_e_v2
   fault_e_nxt       fault_e_xc1       fault_e_xc2       fault_e_yc1
   fault_e_yc2

fault      is POLYGON; with  15 fields,     40 records,     1400 bytes
   Title:
 Created: 1993/09/29;14:51  Modified: 1993/09/29;14:51  Data: 1993/09/29;14:54

 Table attributes:
   Edge table = fault_e   Dimension = 2
 Table has no Keys

 Fields:
   fault_rec         fault_nul         fault_npt         fault_edg
```

```
  fault_xc        fault_yc        fault_are       fault_per
  fault_xmn       fault_ymn       fault_xmx       fault_ymx
  name            tect            tectcol

litho     is  FLAT; with 20 fields,   5040 records,   831808 bytes
  Title: BORE LITHOLOGY INCLUDING ENCODED INFORMATION
Created: 1993/11/02;07:23  Modified: 1994/06/01;16:25  Data: 1994/06/27;17:48

 Table Keys:
l_from        grid          well_no

 Fields:
  litho_rec       litho_nul       *l_from         l_to
  lith1           lith2           sub1            sub2
  gs_no           *grid           *well_no        +HOLEID
  descr           WB              AQUIFER         siz
 &age             &aqseq          &l_patt         +UNIT

f_model   is  LAYER; with 20 fields,  125895 records,   5917065 bytes
  Title: variable size layer table for modflow modelling
Created: 1994/03/21;16:32  Modified: 1994/04/07;13:55  Data: 1994/04/07;16:36

 Table attributes:
  Lower-left X coord:       1200.0  Column size: *     Number:    231
          Y coord:          600.0   Row size: *      Number:    109
  Layers:
    uncon    shallow    medium      deep      vdeep
  Baseline azimuth:  45.00
  Column Definition:
     7  20.0      2  15.0      2  12.5      4  10.0
     1   7.5
     1   5.0
     1   3.5
     1   2.5
     1   1.5
   200   1.0
     1   1.5
     1   2.5
     1   3.5
     1   5.0
     1   7.5
     1  10.0
     1  15.0
     4  20.0
  Row Definition:
     1  20.0      1  15.0      1  10.0      1   7.5
     1   5.0
     1   3.5
     5   2.5
     6   1.5
    75   1.0
     6   1.5
     5   2.5
     1   3.5
     1   5.0
     1   7.5
     1  10.0
     1  15.0
     1  20.0

 Fields:
  f_model_rec     f_model_nul     f_model_row     f_model_col
  f_model_lay     f_model_xc      f_model_csz     f_model_yc
```

f_model_rsz f_model_nam shead bdy
TRANS HCON VCON TOP
BOT STCOF SPCYLD TEMP_LOC

QULO is JOIN; with 2 fields, 0
Table 1: quality to Table 2: location

Fields:
QULO_rec QULO_nul

LOCWOB is JOIN; with 2 fields, 0
Table 1: wobs to Table 2: location

Fields:
LOCWOB_rec LOCWOB_nul

QULOWB is JOIN; with 2 fields, 0
Table 1: QULO to Table 2: wobs

Fields:
QULOWB_rec QULOWB_nul

locli is JOIN; with 2 fields, 0
Table 1: litho to Table 2: location

Fields:
locli_rec locli_nul

Fields not in any table:
TOPO DZDX DZDY LAYER

FIELDS:
location_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 8014
0 bytes stored per value

location_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

grid is an ACTUAL REAL field
Title: MWRC GRID NUMBER
Expected Minimum = 200 Maximum = 1000
4 bytes stored per value. Precision 0

well_no is an ACTUAL INTEGER field
Title: MWRC WELL NUMBER WITHIN A GRID
Expected Minimum = 0 Maximum = 999
2 bytes stored per value

east is an ACTUAL REAL field
Title: 4 DIGIT ABBREVIATED EASTING
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

north is an ACTUAL REAL field
Title: 4 DIGIT ABBREVIATED NORTHING
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

map is an ACTUAL TEXT field
Title: NZMS 260 MAP TITLE
Length = 4 characters, LEFT justified.
4 bytes stored per value

depth is an ACTUAL REAL field
Title: DEPTH OF BORE IN METRES
Expected Minimum = 0 Maximum = 10000
4 bytes stored per value. Precision 0

altit is a MEASURED REAL field
Title: ALTITUDE OF WELLHEAD IN MSL
Expected Minimum = * Maximum = *
9 bytes stored per value. Precision 3
Values: n/a 0.000

swl is a MEASURED REAL field
Title: STATIC WATER LEVEL, IN METRES H2O
Expected Minimum = * Maximum = *
9 bytes stored per value. Precision 1
Values: n/a 0.0

corr is a MEASURED REAL field
Title: DISTANCE IN METRES BETWEEN SWL DATUM AND SURVEYED DATUM
Expected Minimum = * Maximum = *
9 bytes stored per value. Precision 3
Values: N/A NULL

monitor is an ACTUAL TEXT field
Title: MWRC MONITORING RUN CODE, IF BLANK, NO MONITORING
Length = 12 characters, LEFT justified.
12 bytes stored per value

old is an ACTUAL TEXT field
Title: RANGITIKEI WANGANUI CATCHMENT BOARD BORE NUMBER
Length = 7 characters, LEFT justified.
7 bytes stored per value

surname is an ACTUAL TEXT field
Title: SURNAME OF BORE OWNER
Length = 25 characters, LEFT justified.
25 bytes stored per value

init is an ACTUAL TEXT field
Title: INITIALS OF BORE OWNER
Length = 10 characters, LEFT justified.
10 bytes stored per value

alev is a CALCULATED REAL field
Title: AQUIFER ELEVATION IN MSL (ALTIT-DEPTH)

```
   Expected Minimum = *  Maximum = *
   0 bytes stored per value. Precision 0
   Equation: altit depth -

walt       is a  CALCULATED      REAL field
   Title: GROUNDWATER ALTITUDE IN MSL  (ALTIT+SWL)
   Expected Minimum = *  Maximum = *
   0 bytes stored per value. Precision 1
   Equation: NULL altit 0.01 < 7 skip NULL swl == 0.00 3 skip altit swl +

discharge  is an    ACTUAL      REAL field
   Title: WELL TEST DISCHARGE RATE IN M3/DAY
   Expected Minimum = *  Maximum = *
   8 bytes stored per value. Precision 0


drawdown   is an    ACTUAL      REAL field
   Title: DRAWDOWN IN METRES, CORRESPONDING TO DISCHARGE
   Expected Minimum = *  Maximum = *
   8 bytes stored per value. Precision 1


testt      is an    ACTUAL      TEXT field
   Title: TYPE OF WELL TEST S-STEP C-CONSTANT P-PRODUCTION O-OBS.BORE
   Length =  5 characters,  LEFT justified.
   5 bytes stored per value


transmis   is an    ACTUAL      REAL field
   Title: TRANSMISSIVITY IN M2/DAY
   Expected Minimum = *  Maximum = *
   8 bytes stored per value. Precision 0


stor       is an    ACTUAL      REAL field
   Title: STORAGE COEFFICIENT
   Expected Minimum = *  Maximum = *
   8 bytes stored per value. Precision 7


aqtype     is an    ACTUAL      TEXT field
   Title: AQUIFER LITHOLOGY G-GRAVEL S-SAND &-GRAVEL&SAND W-GREYWACKE
   Length = 10 characters,  LEFT justified.
   10 bytes stored per value


spdis      is a  CALCULATED      REAL field
   Title: SPECIFIC DISCHARGE IN M2/DAY (DISCHARGE/DRAWDOWN)
   Expected Minimum = *  Maximum = *
   0 bytes stored per value. Precision 7
   Equation: NULL drawdown 0.01 < 3 skip discharge drawdown /

quality_rec is an  AUTOMATIC    INTEGER field
   Title:
   Expected Minimum = 1  Maximum = 3043
   0 bytes stored per value


quality_nul is an  AUTOMATIC    INTEGER field
   Title:
   Expected Minimum = 0  Maximum = 1
   0 bytes stored per value
```

dateana is an ACTUAL DATE field
Title: DATE OF WATER ANALISYS
Expected Minimum = * Maximum = *
4 bytes stored per value. Precision DAY

ph is an ACTUAL REAL field
Title: PH OF WATER
Expected Minimum = 3.00 Maximum = 10.00
4 bytes stored per value. Precision 2

temp is an ACTUAL REAL field
Title: TEMPERATURE IN DEGREES CELSIUS
Expected Minimum = 0.00 Maximum = 50.00
4 bytes stored per value. Precision 2

cond is an ACTUAL REAL field
Title: CONDUCTIVITY IN mS/m
Expected Minimum = 0.0 Maximum = 10000.0
4 bytes stored per value. Precision 1

ca is an ACTUAL REAL field
Title: CALCIUM CONTENT IN MG/L CACO3
Expected Minimum = 0.00 Maximum = 1000.00
4 bytes stored per value. Precision 2

mg is an ACTUAL REAL field
Title: MAGNESIUM CONTENT IN MG/L CACO3
Expected Minimum = 0.00 Maximum = 1000.00
4 bytes stored per value. Precision 2

tothard is an ACTUAL REAL field
Title: TOTAL HARDNESS IN MG/L CACO3
Expected Minimum = 0.00 Maximum = 2000.00
4 bytes stored per value. Precision 2

fe is a MEASURED REAL field
Title: IRON CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 100.000
5 bytes stored per value. Precision 3
Values: nd 0.010

mn is a MEASURED REAL field
Title: MANGANESE CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 20.000
5 bytes stored per value. Precision 3
Values: nd 0.001

na is an ACTUAL REAL field
Title: SODIUM CONTENT IN MG/L
Expected Minimum = 0.0 Maximum = 999.0
4 bytes stored per value. Precision 1

k is an ACTUAL REAL field
Title: POTASSIUM CONTENT IN MG/L
Expected Minimum = 0.00 Maximum = 999.00
4 bytes stored per value. Precision 2

co is a MEASURED REAL field
Title: CARBONATE ALKALINITY IN MG/L CACO3
Expected Minimum = 0.000 Maximum = 99.000
5 bytes stored per value. Precision 3
Values: nd 0.001

hco is an ACTUAL REAL field
Title: TOTAL ALKALINITY IN MG/L CACO3
Expected Minimum = 0.00 Maximum = 5000.00
4 bytes stored per value. Precision 2

cl is an ACTUAL REAL field
Title: CHLORIDE CONTENT IN MG/L
Expected Minimum = 0.00 Maximum = 5000.00
4 bytes stored per value. Precision 2

so is a MEASURED REAL field
Title: SULPHATE CONTENT IN MG/L
Expected Minimum = 0.00 Maximum = 1000.00
5 bytes stored per value. Precision 2
Values: nd 0.10

f is a MEASURED REAL field
Title: FLUORIDE CONTENT IN MG/L
Expected Minimum = 0.00 Maximum = 50.00
5 bytes stored per value. Precision 2
Values: nd 0.10

co2free is a MEASURED REAL field
Title: FREE CARBON DIOXIDE IN MG/L
Expected Minimum = 0.00 Maximum = 999.00
5 bytes stored per value. Precision 2
Values: nd 0.50

no2 is a MEASURED REAL field
Title: NITRITE-NITROGEN IN MG/L
Expected Minimum = 0.000 Maximum = 50.000
5 bytes stored per value. Precision 3
Values: nd 0.001

no3 is a MEASURED REAL field
Title: NITRATE-NITROGEN IN MG/L
Expected Minimum = 0.000 Maximum = 100.000
5 bytes stored per value. Precision 3
Values: nd 0.010

po4 is a MEASURED REAL field
Title: TOTAL PHOSPHORUS OR DR PHOSPHORUS IN MG/L
Expected Minimum = 0.000 Maximum = 10.000
5 bytes stored per value. Precision 3
Values: nd 0.002

tds is an ACTUAL REAL field

Title: TOTAL DISSOLVED SOLIDS IN MG/L
Expected Minimum = 0.0 Maximum = 9999.0
4 bytes stored per value. Precision 1

nh3 is a MEASURED REAL field
Title: AMMONIACAL NITROGEN IN MG/L
Expected Minimum = 0.00 Maximum = 100.00
5 bytes stored per value. Precision 2
Values: nd 0.02

al is a MEASURED REAL field
Title: ALUMINIUM CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 100.000
5 bytes stored per value. Precision 3
Values: nd 0.001

si is a MEASURED REAL field
Title: SILICA CONTENT IN MG/L
Expected Minimum = * Maximum = 999.00
9 bytes stored per value. Precision 2
Values: nd 0.01

cu is a MEASURED REAL field
Title: COPPER CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 99.000
5 bytes stored per value. Precision 3
Values: nd 0.010

turb is a MEASURED REAL field
Title: TURBIDITY IN N.T.U
Expected Minimum = 0.0 Maximum = 1000.0
5 bytes stored per value. Precision 1
Values: nd 0.0

area is an ACTUAL TEXT field
Title: AREA CODE
Length = 12 characters, LEFT justified.
12 bytes stored per value

QULO_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

QULO_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

date_year is a CALCULATED DATE field
Title: YEAR OF WATER ANALYSIS
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision YEAR
Equation: dateana

temat_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1201
0 bytes stored per value

temat_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

aqcode is an ACTUAL TEXT field
Title: AQUIFER CODE
Length = 6 characters, LEFT justified.
6 bytes stored per value

thick is an ACTUAL REAL field
Title: AQUIFER THICKNESS OR SCREEN LENGHT IN METRES
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 1

se91 is an ACTUAL REAL field
Title: SEPT 1991 SWL MEASUREMENTS IN EITHER BAR OR M
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 1

ju91 is an ACTUAL REAL field
Title: JUNE 1991 SWL DATA IN EITHER BAR (>0) OR M (<0)
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 1

sep91 is a CALCULATED REAL field
Title: SEPT 1991 SWL DATA IN MSL
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 1
Equation: se91 altit + se91 0.0 <= 6 skip pop se91 10.197 * altit +

jun91 is a CALCULATED REAL field
Title: JUNE 1991 SWL DATA IN MSL
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 1
Equation: ju91 altit + ju91 0.0 <= 6 skip pop ju91 10.197 * altit +

mgtoca is a CALCULATED REAL field
Title: MAGNESIUM TO IONIC CA RATIO
Expected Minimum = 0.00 Maximum = 99.00
0 bytes stored per value. Precision 2
Equation: mg ca /

natocl is a CALCULATED REAL field
Title: NA TO CL IONIC RATIO
Expected Minimum = 0.00 Maximum = 999.00
0 bytes stored per value. Precision 2
Equation: ENA ECL / ENA exist 1 skip 0

hardtokna is a CALCULATED REAL field
Title: CA + MG TO NA + K IONIC RATIO
Expected Minimum = 0.0 Maximum = 99999.0
0 bytes stored per value. Precision 1
Equation: ECAMG ENAK / ECAMG exist 1 skip 0

natok is a CALCULATED REAL field
Title: NA TO K IONIC RATIO

Expected Minimum = 0.0 Maximum = 9999.0
0 bytes stored per value. Precision 1
Equation: ENA EK / ENA exist 1 skip 0

slot is an ACTUAL REAL field
Title: SCREEN SLOT SIZE IN 1/1000 INCHES
Expected Minimum = 0 Maximum = 99999
4 bytes stored per value. Precision 0

ABS is an ACTUAL REAL field
Title: ABSORBANCE AT 270 nm
Expected Minimum = 0.000 Maximum = 99.000
4 bytes stored per value. Precision 3

FCOL is a MEASURED REAL field
Title: FAECAL COLIFORMS IN 1/100 ML
Expected Minimum = 0 Maximum = 9999
5 bytes stored per value. Precision 0
Values: nd 0 heaps 1000

CODE is an ACTUAL TEXT field
Title: CODE TO SHOW IF WATER QUALITY IS MONITORED OR NOT
Length = 3 characters, LEFT justified.
3 bytes stored per value

ma92 is an ACTUAL REAL field
Title: MARCH 1992 SWL DATA IN EITHER BAR OR M
Expected Minimum = -20.000 Maximum = 5.000
4 bytes stored per value. Precision 3

MAR92 is a CALCULATED REAL field
Title: MARCH 1992 SWL DATA IN MSL
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 1
Equation: ma92 altit + ma92 0.0 <= 6 skip pop ma92 10.197 * altit +

TOPO is an ACTUAL REAL field
Title:
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

DZDX is an ACTUAL REAL field
Title:
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

DZDY is an ACTUAL REAL field
Title:
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

usage_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1642
0 bytes stored per value

usage_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

meast is an ACTUAL REAL field
Title: 7 DIGIT FULL METRIC EASTING
Expected Minimum = 0 Maximum = 4000000
8 bytes stored per value. Precision 0

mnorth is an ACTUAL REAL field
Title: 7 DIGIT FULL METRIC NORTHING
Expected Minimum = 0 Maximum = 7000000
8 bytes stored per value. Precision 0

userate is an ACTUAL REAL field
Title: WATER USE, IN M3/DAY, IMPORTED FROM CONMAN
Expected Minimum = 0 Maximum = 99999
4 bytes stored per value. Precision 0

eastc is a CALCULATED REAL field
Title: CALCULATED 4 DIGIT EASTING
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0
Equation: meast 100 / 26000 -

northc is a CALCULATED REAL field
Title: CALCULATED 4 DIGIT NORTHING
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0
Equation: mnorth 100 / 60000 -

type is an ACTUAL TEXT field
Title: TYPE OF WATER PERMIT IMPORTED FROM CONMAN "SUBTYPE"
Length = 15 characters, LEFT justified.
15 bytes stored per value

permit is an ACTUAL TEXT field
Title: WATER PERMIT NUMBER IMPORTED FROM CONMAN
Length = 12 characters, LEFT justified.
12 bytes stored per value

totcol is a MEASURED INTEGER field
Title: TOTAL COLIFORMS IN 1/100 ML
Expected Minimum = 0 Maximum = 100000
5 bytes stored per value
Values: nd NULL

LOCALITY is an ACTUAL TEXT field
Title: LOCALITY CODE WITHIN AREA, IE BUNNYTHORPE WITHIN MANAWATU
Length = 20 characters, LEFT justified.
20 bytes stored per value

region is an ACTUAL TEXT field
Title: MWRC IF WITHIN MWRC REGION; TO TEST LOCATE WITHIN MWRC.DIG
Length = 5 characters, LEFT justified.

5 bytes stored per value

concol is a MEASURED INTEGER field
Title: CONFIRMED COLIFORMS IN 1/100 ML
Expected Minimum = 0 Maximum = 100000
5 bytes stored per value
Values: nd 0 heaps 1000

BOD5 is a MEASURED REAL field
Title: FIVE DAY BIOLOGICAL OXIGEN DEMAND
Expected Minimum = 0.0 Maximum = 10000.0
5 bytes stored per value. Precision 1
Values: nd 0.1

SUSP is a MEASURED REAL field
Title: SUSPENDED SOLIDS IN MG/L
Expected Minimum = 0.00 Maximum = 10000.00
9 bytes stored per value. Precision 2
Values: nd 0.10

qcount is an ACTUAL REAL field
Title: COUNTER SET TO 1 IF QUALITY RECORD IS THE LATEST FOR SITE
Expected Minimum = 0 Maximum = 999
4 bytes stored per value. Precision 0

wobs_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1618
0 bytes stored per value

wobs_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

WLOBS is an ACTUAL REAL field
Title: OBSERVED WATER LEVEL IN M
Expected Minimum = -999.0 Maximum = 999.0
4 bytes stored per value. Precision 1

boron is a MEASURED REAL field
Title: BORON CONTENT IN MG/L
Expected Minimum = 0.00 Maximum = 100.00
5 bytes stored per value. Precision 2
Values: nd 0.01

decol is a CALCULATED INTEGER field
Title: BORE DEPTH COLOUR AS ON BORE MAPS
Expected Minimum = 0 Maximum = 20
0 bytes stored per value
Equation: 1 depth 150 >= 25 skip 5 depth 107 >= 19 skip 4 depth 70 >= 13
skip 2 depth 50 >= 7 skip 3 depth 21 >= 1 skip 7

mg_id is an ACTUAL TEXT field
Title: NZMS 260 MAP NAME SUCH AS S24
Length = 15 characters, LEFT justified.
15 bytes stored per value

mg_col is an ACTUAL INTEGER field
Title: MAP COLOUR
Expected Minimum = 0 Maximum = 200
2 bytes stored per value

mg_style is an ACTUAL INTEGER field
Title: MAP STYLE
Expected Minimum = 0 Maximum = 200
2 bytes stored per value

JUL92 is an ACTUAL REAL field
Title: JULY 1992 SWL DATA IN EITHER BAR OR M
Expected Minimum = -100.000 Maximum = 100.000
8 bytes stored per value. Precision 3

JULY92 is a CALCULATED REAL field
Title: JULY 1992 SWL DATA IN MSL
Expected Minimum = 0.0 Maximum = 1000.0
0 bytes stored per value. Precision 1
Equation: JUL92 altit + JUL92 0.0 <= 6 skip pop JUL92 10.197 * altit +

where is an ACTUAL TEXT field
Title: WHERE TO DISCHARGE IE LAND/WATER
Length = 8 characters, LEFT justified.
8 bytes stored per value

DISRATE is an ACTUAL REAL field
Title: DISRATE IN M3/DAY IMPORTED FROM CONMAN
Expected Minimum = * Maximum = *
8 bytes stored per value. Precision 0

DISCH_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 2393
0 bytes stored per value

DISCH_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

cr is a MEASURED REAL field
Title: CHROMIUM CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 100.000
5 bytes stored per value. Precision 3
Values: nd 0.010

as is a MEASURED REAL field
Title: ARSENIC CONTENT IN MG/L
Expected Minimum = 0.000 Maximum = 100.000
5 bytes stored per value. Precision 3
Values: nd 0.001

resist_rec is an AUTOMATIC INTEGER field
Title:

Expected Minimum = 1 Maximum = 2165
0 bytes stored per value

resist_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

sounding is an ACTUAL TEXT field
Title: NAME OF RESISTIVITY SOUNDING
Length = 10 characters, LEFT justified.
10 bytes stored per value

scode is an ACTUAL TEXT field
Title: RESISTIVITY SOUNDING CODE
Length = 10 characters, LEFT justified.
10 bytes stored per value

abhalf is an ACTUAL REAL field
Title: AB/2 DISTANCE IN M
Expected Minimum = 0.0 Maximum = 9999.0
4 bytes stored per value. Precision 1

ra is an ACTUAL REAL field
Title: APPARENT RESISTIVITY IN OHMM
Expected Minimum = 0.0 Maximum = 99999.0
4 bytes stored per value. Precision 1

phs is an ACTUAL REAL field
Title: SATURATED PH
Expected Minimum = 0.00 Maximum = 14.00
4 bytes stored per value. Precision 2

colour is an ACTUAL REAL field
Title:
Expected Minimum = 0 Maximum = 999
4 bytes stored per value. Precision 0

Zn is an ACTUAL REAL field
Title: ZINK CONTENT IN MG/L
Expected Minimum = 0.0000 Maximum = 999.0000
8 bytes stored per value. Precision 4

flow is an ACTUAL INTEGER field
Title: FLOW MEASURED IN M3/MIN
Expected Minimum = 0 Maximum = 20000
2 bytes stored per value

FEB93 is an ACTUAL REAL field
Title: FEBRUARY 1993 SWL DATA IN EITHER BARS OR M
Expected Minimum = -99.0 Maximum = 99.0
4 bytes stored per value. Precision 1

FEBR93 is a CALCULATED REAL field
Title: FEBRUARY 1993 SWL DATA IN MSL
Expected Minimum = 0.0 Maximum = 999.0
0 bytes stored per value. Precision 1
Equation: FEB93 altit + FEB93 0.0 <= 6 skip pop FEB93 10.197 * altit +

HOLEID is a CALCULATED REAL field
Title: CALCULATED ID FIELD
Expected Minimum = 0 Maximum = 99999999
0 bytes stored per value. Precision 0
Equation: grid 1000 * well_no +

DDE is a CALCULATED REAL field
Title: CALCULATED DEPTH FIELD USED FOR CROSS SECTION DOWNHOLE LAB
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 2
Equation: depth 0.5 -

grid_id is an ACTUAL INTEGER field
Title: GRID_ID USED ON NZMS 260 BORE MAPS
Expected Minimum = 0 Maximum = 999
2 bytes stored per value

grid_col is an ACTUAL INTEGER field
Title: GRID COLOUR
Expected Minimum = 0 Maximum = 256
2 bytes stored per value

grid_sty is an ACTUAL INTEGER field
Title: GRID STYLE
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

HOLDER is an ACTUAL TEXT field
Title: NAME OF PERMIT HOLDER IMPORTED FROM CONMAN
Length = 24 characters, LEFT justified.
24 bytes stored per value

recstart is an ACTUAL DATE field
Title: MONITORING RECORD START
Expected Minimum = 1900/01/01 Maximum = 2000/01/01
4 bytes stored per value. Precision DAY

RECSTOP is an ACTUAL DATE field
Title: MONITORING RECORD END DATE
Expected Minimum = 1900/01/01 Maximum = 2000/01/01
4 bytes stored per value. Precision DAY

RECLEN is a CALCULATED REAL field
Title: CALCULATED RECORD LENGTH
Expected Minimum = 0.0 Maximum = 100.0
0 bytes stored per value. Precision 1
Equation: RECSTOP recstart - 365 /

INTCOL is a CALCULATED INTEGER field
Title: CALCULATED DEPTH COLOUR-CODE

Expected Minimum = 0 Maximum = 100
0 bytes stored per value
Equation: 8 depth 70 > 6 skip depth 10 / int 1 +

mapgrids_v_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1864
0 bytes stored per value

mapgrids_v_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

mapgrids_v_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 1
8 bytes stored per value. Precision 0

mapgrids_v_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 1
8 bytes stored per value. Precision 0

mapgrids_e_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1930
0 bytes stored per value

mapgrids_e_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

mapgrids_e_v1 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

mapgrids_e_v2 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

mapgrids_e_nxt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

mapgrids_e_xc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_e_xc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_e_yc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_e_yc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1028
0 bytes stored per value

mapgrids_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

mapgrids_npt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

mapgrids_edg is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

mapgrids_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_are is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_per is an AUTOMATIC REAL field
Title:

Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_xmn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_ymn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_xmx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

mapgrids_ymx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_v_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1276
0 bytes stored per value

grids_v_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

grids_v_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 1
8 bytes stored per value. Precision 0

grids_v_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 1
8 bytes stored per value. Precision 0

grids_e_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1798
0 bytes stored per value

grids_e_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

grids_e_v1 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

grids_e_v2 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

grids_e_nxt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

grids_e_xc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_e_xc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_e_yc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_e_yc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1138
0 bytes stored per value

grids_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

grids_npt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

grids_edg is an AUTOMATIC INTEGER field
Title:

Expected Minimum = 1 Maximum = *
4 bytes stored per value

grids_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_are is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_per is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_xmn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_ymn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_xmx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

grids_ymx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

GCLASS is an ACTUAL INTEGER field
Title: 1:BEDROCK 2:COMPACTED GRAVEL 3:>20M MIXED 4:>10M SOFT
Expected Minimum = 0 Maximum = 99
1 bytes stored per value

blog is an ACTUAL TEXT field
Title: BORELOG AVAILABLE OR NOT
Length = 1 characters, LEFT justified.
1 bytes stored per value

AREACOL is an ACTUAL INTEGER field
Title: COLOUR CODE FOR AREAS
Expected Minimum = 0 Maximum = *
4 bytes stored per value

GRAVITY_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1378
0 bytes stored per value

GRAVITY_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

RESID is an ACTUAL REAL field
Title: RESIDUAL BOUGUER ANOMALY IN microN/kg
Expected Minimum = -10000.00 Maximum = 10000.00
8 bytes stored per value. Precision 2

BOUGER is an ACTUAL REAL field
Title: total bouger anomaly in microN/kg
Expected Minimum = -10000.00 Maximum = 10000.00
8 bytes stored per value. Precision 2

FEAST is a CALCULATED INTEGER field
Title: calculated 7 digit easting
Expected Minimum = 2000000 Maximum = 9999999
0 bytes stored per value
Equation: east 100 * 2600000 +

FNORTH is a CALCULATED INTEGER field
Title: calculated 7 digit north
Expected Minimum = 2000000 Maximum = 9999999
0 bytes stored per value
Equation: north 100 * 6000000 +

ECA is a CALCULATED REAL field
Title: CA IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ca 0.01998 * ca exist 1 skip 0

EMG is a CALCULATED REAL field
Title: MG IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: mg 0.01998 * mg exist 1 skip 0

ECAMG is a CALCULATED REAL field
Title: CA + MG IN MEQUIVALENT UNIST
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ECA EMG + ECA exist 5 skip EMG exist 1 skip 0

EK is a CALCULATED REAL field
Title: K IN MEQUIVALENT UNITS

Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: k 0.2557 * k exist 1 skip 0

ENA is a CALCULATED REAL field
Title: NA IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: na 0.0435 * na exist 1 skip 0

ENAK is a CALCULATED REAL field
Title: NA + K IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ENA EK + ENA exist 5 skip EK exist 1 skip 0

CATSUM is a CALCULATED REAL field
Title: calculated cation sum in mequivalent units
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ENAK ECAMG + ENAK exist 5 skip ECAMG exist 1 skip 0

PCA is a CALCULATED REAL field
Title: CA AS PERCENTAGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ECA CATSUM /

PMG is a CALCULATED REAL field
Title: MG AS PERCENTAGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: EMG CATSUM /

PCAMG is a CALCULATED REAL field
Title: CA + MG AS PERCENATGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ECAMG CATSUM /

PNA is a CALCULATED REAL field
Title: NA AS PERECNTAGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ENA CATSUM /

PK is a CALCULATED REAL field
Title: K AS PERECNTAGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: EK CATSUM /

PNAK is a CALCULATED REAL field
Title: NA + K AS PERCENTAGE OF CATIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ENAK CATSUM /

EHCO is a CALCULATED REAL field
Title: ALKALINITY IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: hco 0.01998 * hco exist 1 skip 0

ECL is a CALCULATED REAL field
Title: CL IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: cl 0.02821 * cl exist 1 skip 0

ESO is a CALCULATED REAL field
Title: SO IN MEQUIVALENT UNITS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: so 0.02083 * so exist 1 skip 0

ANSUM is a CALCULATED REAL field
Title: CALCULATED ANION SUM IN MEQUIVALENTS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: EHCO ECL + ESO + EHCO exist 9 skip ECL exist 5 skip ESO exist 1
skip 0

PCL is a CALCULATED REAL field
Title: CL AS PERCENTAGE OF ANIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ECL ANSUM / ECL exist 1 skip 0

PSO is a CALCULATED REAL field
Title: SO AS PERCENTAGE OF ANIONS
Expected Minimum = 0.00 Maximum = 99999.00
0 bytes stored per value. Precision 2
Equation: ESO ANSUM / ESO exist 1 skip 0

PHCO is a CALCULATED REAL field
Title: ALAKLINITY AS PERCENTAGE OF ANIONS
Expected Minimum = 0.00 Maximum = 1.00
0 bytes stored per value. Precision 2
Equation: EHCO ANSUM / hco exist 1 skip 0

sar is a CALCULATED REAL field
Title: CALCULATED SODIUM ADSORPTION RATIO
Expected Minimum = 0 Maximum = 9999
0 bytes stored per value. Precision 0
Equation: ENA ECA EMG + 2 / 0.5 ^ / ENA exist 1 skip 0

WHAKA_MOD is an ACTUAL TEXT field
Title: IF THE BORE IS WITHIN THE WHAKA MODEL AREA SET TO "Y"
Length = 1 characters, LEFT justified.
1 bytes stored per value

W_ROW is an ACTUAL REAL field
Title: WHAK MODEL ROW NUMBER
Expected Minimum = 0.00 Maximum = 99999.00
8 bytes stored per value. Precision 2

W_COL is an ACTUAL REAL field
Title: WHAKA MODEL COLUMN NUMBER
Expected Minimum = 0.00 Maximum = 99999.00
8 bytes stored per value. Precision 2

LAYER is an ACTUAL INTEGER field

Title: LAYER NAME, CORRESPONDS TO W_MODEL LAYER TABLE
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

source is an ACTUAL TEXT field
Title: SOURCE OF INFORMATION BLOG, PHOTOMAP ETC
Length = 6 characters, LEFT justified.
6 bytes stored per value

LOCWOB_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

LOCWOB_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

QULOWB_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

QULOWB_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

lay_type_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1004
0 bytes stored per value

lay_type_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

lt is an ACTUAL INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

antrpy is an ACTUAL REAL field
Title:
Expected Minimum = 0.00 Maximum = 1.00
4 bytes stored per value. Precision 2

WL is an ACTUAL REAL field
Title:
Expected Minimum = 0.0 Maximum = 999.0
4 bytes stored per value. Precision 1

WHAQCOL is an ACTUAL INTEGER field
Title: COLOUR FOR WHAKA_MOD BORES
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

SWCODE is a CALCULATED INTEGER field
Title: SOLOUR CODE FOR SWL
Expected Minimum = 0 Maximum = 9
0 bytes stored per value
Equation: 0 swl NULL == 23 skip 1 swl 10 >= 19 skip 2 swl 2.5 >= 13 skip 3
swl -5 >= 7 skip 6 swl -15 >= 1 skip 7

DECODE is a CALCULATED INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
0 bytes stored per value
Equation: 1 depth 100 >= 19 skip 2 depth 50 >= 13 skip 3 depth 30 >= 7 skip
6 depth 15 >= 1 skip 4

AQNUMB is an ACTUAL INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

tect is an ACTUAL TEXT field
Title:
Length = 9 characters, LEFT justified.
9 bytes stored per value

name is an ACTUAL TEXT field
Title:
Length = 20 characters, LEFT justified.
20 bytes stored per value

tectcol is an ACTUAL INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

fault_v_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1138
0 bytes stored per value

fault_v_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

fault_v_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 99999999
8 bytes stored per value. Precision 0

fault_v_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = 0 Maximum = 99999999
8 bytes stored per value. Precision 0

fault_e_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 1098
0 bytes stored per value

fault_e_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

fault_e_v1 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

fault_e_v2 is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

fault_e_nxt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

fault_e_xc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_e_xc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_e_yc1 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_e_yc2 is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_rec is an AUTOMATIC INTEGER field
Title:

Expected Minimum = 1 Maximum = 1040
0 bytes stored per value

fault_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

fault_npt is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

fault_edg is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
4 bytes stored per value

fault_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_are is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_per is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_xmn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_ymn is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_xmx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

fault_ymx is an AUTOMATIC REAL field
Title:
Expected Minimum = * Maximum = *
0 bytes stored per value. Precision 0

DRILLER is an ACTUAL TEXT field
Title: NAME OF DRILLING COMPANY R-RICHARDSON N-N. WEBB W-WANGANUI
Length = 12 characters, LEFT justified.
12 bytes stored per value

DIA is an ACTUAL REAL field
Title: BORE DIAMETER IN INCHES
Expected Minimum = 0.0 Maximum = 999.0
4 bytes stored per value. Precision 1

DATE_DRILL is an ACTUAL DATE field
Title: DATE OF DRILLING
Expected Minimum = 1900/01/01 Maximum = 2050/01/01
4 bytes stored per value. Precision DAY

litho_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 5096
0 bytes stored per value

litho_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

l_from is an ACTUAL REAL field
Title: TOP OF STRATA IN METRES BELOW DATUM
Expected Minimum = 0.00 Maximum = 999.00
4 bytes stored per value. Precision 2

l_to is an ACTUAL REAL field
Title: BOTTOM OF STRATA IN METRES BELOW DATUM
Expected Minimum = 0.00 Maximum = 999.00
4 bytes stored per value. Precision 2

lith1 is an ACTUAL TEXT field
Title: MAIN LITHOLOGY G:GRAVEL,S:SAND,Z:SILT,C:CLAY,l;LIMESTONE
Length = 10 characters, LEFT justified.
10 bytes stored per value

lith2 is an ACTUAL TEXT field
Title: MINOR LITHOLOGY CODED AS LITH1
Length = 10 characters, LEFT justified.
10 bytes stored per value

sub1 is an ACTUAL TEXT field
Title: DESCRIPTIVE LITHOLOGY, IE S IF SANDY ...

Length = 10 characters, LEFT justified.
10 bytes stored per value

sub2 is an ACTUAL TEXT field
Title:
Length = 10 characters, LEFT justified.
10 bytes stored per value

gs_no is an ACTUAL TEXT field
Title:
Length = 10 characters, LEFT justified.
10 bytes stored per value

LOCD is an ACTUAL INTEGER field
Title: INTEGER FIELD TO AID LOCATING WITHIN POLYGONS
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

ECOCCI is a MEASURED INTEGER field
Title:
Expected Minimum = 0 Maximum = 999999
5 bytes stored per value
Values: nd 0

f_model_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 126895
0 bytes stored per value

f_model_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

f_model_row is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 109
0 bytes stored per value

f_model_col is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 109
0 bytes stored per value

f_model_lay is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = 5
0 bytes stored per value

f_model_xc is an AUTOMATIC REAL field
Title:
Expected Minimum = 1200.0 Maximum = 1200.0
0 bytes stored per value. Precision 1

f_model_csz is an AUTOMATIC REAL field
Title:
Expected Minimum = 20.0 Maximum = 20.0
0 bytes stored per value. Precision 1

f_model_yc is an AUTOMATIC REAL field
Title:
Expected Minimum = 600.0 Maximum = 3654.7
0 bytes stored per value. Precision 1

f_model_rsz is an AUTOMATIC REAL field
Title:
Expected Minimum = 20.0 Maximum = 20.0
0 bytes stored per value. Precision 1

f_model_nam is an AUTOMATIC TEXT field
Title:
Length = 11 characters, LEFT justified.
0 bytes stored per value

shead is an ACTUAL REAL field
Title:
Expected Minimum = -99.0 Maximum = 999.0
4 bytes stored per value. Precision 1

COL is a CALCULATED INTEGER field
Title:
Expected Minimum = 0 Maximum = 999
0 bytes stored per value
Equation: W_COL dup int - 0.5 >= 1 * W_COL +

ROW is a CALCULATED INTEGER field
Title:
Expected Minimum = 0 Maximum = 999
0 bytes stored per value
Equation: W_ROW dup int - 0.5 >= 1 * W_ROW +

descr is an ACTUAL TEXT field
Title:
Length = 80 characters, LEFT justified.
80 bytes stored per value

bdy is an ACTUAL INTEGER field
Title:
Expected Minimum = -9 Maximum = 9
1 bytes stored per value

td_cm is a CALCULATED REAL field
Title:
Expected Minimum = 0.0 Maximum = 999999.0
0 bytes stored per value. Precision 1
Equation: depth 100 *

TRANS is an ACTUAL REAL field
Title:

Expected Minimum = 0 Maximum = 99999
4 bytes stored per value. Precision 0

HCON is an ACTUAL REAL field
Title:
Expected Minimum = 0 Maximum = 99999
4 bytes stored per value. Precision 0

VCON is an ACTUAL REAL field
Title:
Expected Minimum = 0.00000 Maximum = 999.00000
8 bytes stored per value. Precision 5

TOP is an ACTUAL REAL field
Title:
Expected Minimum = -999.0 Maximum = 999.0
4 bytes stored per value. Precision 1

BOT is an ACTUAL REAL field
Title:
Expected Minimum = -999.0 Maximum = 999.0
4 bytes stored per value. Precision 1

STCOF is an ACTUAL REAL field
Title:
Expected Minimum = 0.000000 Maximum = 1.000000
8 bytes stored per value. Precision 6

SPCYLD is an ACTUAL REAL field
Title:
Expected Minimum = 0.000000 Maximum = 1.000000
8 bytes stored per value. Precision 6

TEMP_LOC is an ACTUAL INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

WB is an ACTUAL TEXT field
Title: WATER BEARING OR NOT (Y/N)
Length = 1 characters, LEFT justified.
1 bytes stored per value

AQUIFER is an ACTUAL TEXT field
Title: AQUIFER NAME IF WATER BEARING
Length = 7 characters, LEFT justified.
7 bytes stored per value

locli_rec is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 1 Maximum = *
0 bytes stored per value

locli_nul is an AUTOMATIC INTEGER field
Title:
Expected Minimum = 0 Maximum = 1
0 bytes stored per value

siz is an ACTUAL INTEGER field
Title: 0 IF NON-AQUIFER, 1 IF AQUIFER
Expected Minimum = 0 Maximum = 9
1 bytes stored per value

age is a MEASURED INTEGER field
Title: AGE OF STRATA IN YEARS BP (RADIOCARBON U ETC DATING)
Expected Minimum = 0 Maximum = 999999
5 bytes stored per value
Values: old 36000

aqseq is a MEASURED INTEGER field
Title: 0 IF NON-AQUIFER, OTHER FOE DIFFERENT AQUIFERS
Expected Minimum = 0 Maximum = 9
2 bytes stored per value
Values: uncon 7 seeon 2 medeon 4 deepeon 3 vdeep 6

l_patt is a MEASURED INTEGER field
Title:
Expected Minimum = 0 Maximum = 999
3 bytes stored per value
Values: G 74 GS 75 SG 63 GZ 82 GC 83 CG 83 ZG 82 S 79 ZS 84 SZ 84 Z 12 ZC
86 CZ 86 SC 85 CS 85 C 41 L 21 O 44 P 44 B 18 V 26 GO 91 SO 92 ZO
93 CO 94 PO 44 GP 91 SP 92 ZP 93 CP 94 OP 44 OG 91 OS 91 OZ 93 OC
94 PG 91 PS 92 PZ 93 PC 94 LG 101 LS 102 LZ 103 LC 104 LO 105 GL
101 SL 102 ZL 103 CL 104 OL 105 VG 111 ...

UNIT is a CALCULATED INTEGER field
Title:
Expected Minimum = 0 Maximum = 9
0 bytes stored per value
Equation: 1

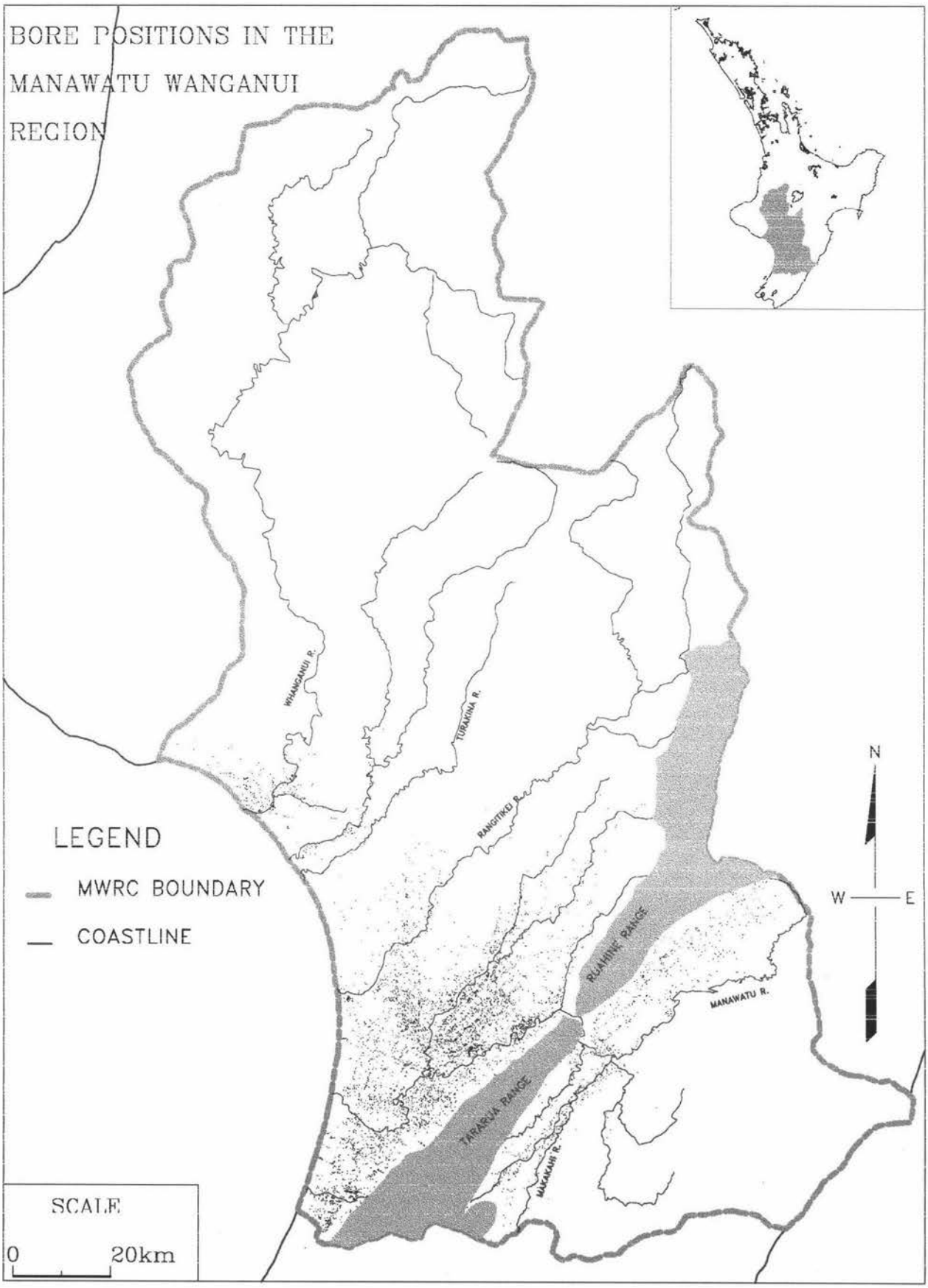
TRMEAN is an ACTUAL REAL field
Title: TRITIUM RATIO MEAN VALUE
Expected Minimum = -9.000 Maximum = 9.000
4 bytes stored per value. Precision 3

TRDEV is an ACTUAL REAL field
Title: TRITIUM RATIO DEVIATION
Expected Minimum = -9.000 Maximum = 9.000
4 bytes stored per value. Precision 3

D18O is an ACTUAL REAL field
Title: OXYGEN 18 - DIFFRENECE TO VIENNA SMOW
Expected Minimum = -99.00 Maximum = 99.00
4 bytes stored per value. Precision 2

Appendix F

Figure F.1 Map: Bore Positions in the Manawatu Wanganui Region (by courtesy of Gabor Bekesi, Manawatu-Wanganui Regional Council)


BORE POSITIONS IN THE
MANAWATU WANGANUI
REGION
WHANGANUI R.
TURAKINA R.
RANGITIKEI R.
RUAHINE RANGE
MANAWATU R.
TARARUA RANGE
MAKAKAHI R.
N
W
E
LEGEND
MWRC BOUNDARY
COASTLINE
SCALE
0
20km