1

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services Branch

Direction des acquisitions et
des services bibliographiques

395 Wellington Street
Ottawa, Ontario
K1A 0N4

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file - Votre référence

Our file - Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

TMDAS
THEMATIC MAP DESIGN ADVISORY SYSTEM
FOR GEOGRAPHICAL INFORMATION SYSTEMS
AND ELECTRONIC MAPPING SYSTEMS

By Jun Yang

**A thesis submitted to
The faculty of Graduate Studies and Research
in partial fulfilment for the degree of**

Master of Arts

Department of Geography

**Carleton University
Ottawa, Ontario
August 3, 1993
© copyright
1993, Jun Yang**

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-89841-0

Canada

Name

JUN YANG

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

Geography
SUBJECT TERM

0366

SUBJECT CODE

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS

Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION

General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0524
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523

Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS

Language 0679
General 0289
Ancient 0290
Linguistics 0291
Modern
Literature
General 0401
Classical 0294
Comparative 0295
Medieval 0297
Modern 0298
African 0316
American 0591
Asian 0305
Canadian (English) 0352
Canadian (French) 0355
English 0593
Germanic 0311
Latin American 0312
Middle Eastern 0315
Romance 0313
Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY

Philosophy 0422
Religion
General 0318
Biblical Studies 0321
Clergy 0319
History of 0320
Philosophy of 0322
Theology 0469

SOCIAL SCIENCES

American Studies 0323
Anthropology
Archaeology 0324
Cultural 0326
Physical 0327
Business Administration
General 0310
Accounting 0272
Banking 0770
Management 0454
Marketing 0338
Canadian Studies 0385
Economics
General 0501
Agricultural 0503
Commerce-Business 0505
Finance 0508
History 0509
Labor 0510
Theory 0511
Folklore 0358
Geography 0366
Gerontology 0351
History
General 0578

Ancient 0579
Medieval 0581
Modern 0587
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science
General 0615
International Law and Relations 0616
Public Administration 0617
Recreation 0814
Social Work 0452
Sociology
General 0626
Criminology and Penology 0627
Demography 0938
Ethnic and Racial Studies 0631
Individual and Family Studies 0628
Industrial and Labor Relations 0629
Public and Social Welfare 0630
Social Structure and Development 0700
Theory and Methods 0344
Transportation 0709
Urban and Regional Planning 0999
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture
General 0473
Agronomy 0285
Animal Culture and Nutrition 0475
Animal Pathology 0476
Food Science and Technology 0359
Forestry and Wildlife 0478
Plant Culture 0479
Plant Pathology 0480
Plant Physiology 0817
Range Management 0777
Wood Technology 0746
Biology
General 0306
Anatomy 0287
Biostatistics 0308
Botany 0309
Cell 0379
Ecology 0329
Entomology 0353
Genetics 0369
Limnology 0793
Microbiology 0410
Molecular 0307
Neuroscience 0317
Oceanography 0416
Physiology 0433
Radiation 0821
Veterinary Science 0778
Zoology 0472
Biophysics
General 0786
Medical 0760

EARTH SCIENCES

Biogeochemistry 0425
Geochemistry 0996

Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES

Environmental Sciences 0768
Health Sciences
General 0566
Audiology 0300
Chemotherapy 0992
Dentistry 0567
Education 0350
Hospital Management 0769
Human Development 0758
Immunology 0982
Medicine and Surgery 0564
Mental Health 0247
Nursing 0569
Nutrition 0570
Obstetrics and Gynecology 0380
Occupational Health and Therapy 0354
Ophthalmology 0381
Pathology 0571
Pharmacology 0419
Pharmacy 0572
Physical Therapy 0382
Public Health 0573
Radiology 0574
Recreation 0575

Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences
Chemistry
General 0485
Agricultural 0749
Analytical 0386
Biochemistry 0487
Inorganic 0488
Nuclear 0738
Organic 0490
Pharmaceutical 0491
Physical 0494
Polymer 0495
Radiation 0754
Mathematics 0405
Physics
General 0605
Acoustics 0986
Astronomy and Astrophysics 0606
Atmospheric Science 0608
Atomic 0748
Electronics and Electricity 0607
Elementary Particles and High Energy 0798
Fluid and Plasma 0759
Molecular 0609
Nuclear 0610
Optics 0752
Radiation 0756
Solid State 0611
Statistics 0463
Applied Sciences
Applied Mechanics 0346
Computer Science 0984

Engineering
General 0537
Aerospace 0538
Agricultural 0539
Automotive 0540
Biomedical 0541
Chemical 0542
Civil 0543
Electronics and Electrical 0544
Heat and Thermodynamics 0348
Hydraulic 0545
Industrial 0546
Marine 0547
Materials Science 0794
Mechanical 0548
Metallurgy 0743
Mining 0551
Nuclear 0552
Packaging 0549
Petroleum 0765
Sanitary and Municipal 0554
System Science 0790
Geotechnology 0428
Operations Research 0796
Plastics Technology 0795
Textile Technology 0994

PSYCHOLOGY

General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

**The undersigned recommend to the Faculty of Graduate Studies
and Research acceptance of the thesis**

TMDAS

THEMATIC MAP DESIGN ADVISORY SYSTEM

**FOR GEOGRAPHICAL INFORMATION SYSTEMS
AND ELECTRONIC MAPPING SYSTEMS**

Submitted by Jun Yang

**in partial fulfilment of the requirements for
the degree of Master of Arts**

Thesis Supervisors

Chair, Department of Geography

Carleton University

Sept. 9, 1993

ABSTRACT

The research focuses on the development of a Thematic Map Design Advisory System (TMDAS) for geographic information systems and electronic mapping systems. The objectives of the project are to extract the inherent knowledge of thematic map design, and to provide this knowledge in an automated mapping environment, through the use of expert system technology, in order to safeguard against the creation of misleading maps. The research was carried out as a knowledge engineering project, in which the concept design was developed for an expert system in thematic map design; the knowledge was codified, further developed, and transformed into formal knowledge; this knowledge was structured into modules; knowledge representation schemes for different types of knowledge were studied and applied; an expert system shell, named J.shell, was developed; the TMDAS knowledge base, with about 500 knowledge entries, was implemented in the J.shell format; and, finally, preliminary case tests and analysis were accomplished. Each of these aspects is discussed in detail in the thesis. Recommendations for further development of the system are also proposed.

ACKNOWLEDGMENTS

I would first like to express my deep appreciation to my supervisor, Dr. D. R. F. Taylor, for his advice, warm support and encouragement throughout the M.A. program.

I especially wish to thank Dr. E. M. Siekierska, the Manager of the Innovation & Technology Service (ITS), Canada Center for Mapping (CCM), Department of Natural Resources, and the adjunct advisor of the M. A. thesis, for her constructive criticism, detailed discussions, and kind encouragement. Eva, I shall remember both Mexico City and Winnipeg.

Grateful thanks also go to the Department of Geography, Carleton University, where I have been involved with an active, friendly group of individuals. These two years of study will be one of the most memorable periods in my life. Special thanks to Dr. M. Fox, Dr. I. Wallace, and Ms. H. Anderson for their help through the entire M.A. program. I also like to thank Dr. D. King and Mr. S. Prashker for the many interesting discussions I had while working with them. I would like to express my special thanks to Ms. C. Earl, for reviewing the thesis proposal, the knowledge acquisition tables, and the prototype of the system, and for her valuable opinions.

I also wish to thank ITS, CCM, where I started the thesis work as part of a COSEP job, for providing partial funds and facility support. I would like to thank the staff of ITS who were always willing to help and were generous with their time. In particular, I like to thank Dr. C. Armenakis, and Dr. D. Richardson for their kindly help during the initial stage of the project. Special thanks to Dr. L. A. Lapalme-Roy for proof-reading the thesis, as well as for constructive recommendations.

I wish to thank many of my friends who have kept me from being lonely, even in a new country. Special thanks to my friends Aining and Shabai for their care, friendship and help in many aspects of my life.

My thanks are extended to my family members, my mother and my sisters, who have continuously given their understanding and support in my life. In particular, I would like to thank my parents in law, who have been taking care of my little daughter on the other side of the Earth. Without their help, I would not be able to study in Canada. I also extend appreciation to my sweet daughter, Qian, for her "consciousness cooperation".

Last, but not least, I would like to thank my husband Zhenwu, for so many evenings when he sat by my side helping to debug my system, for his kindly support, and for many other things we both know.

TABLE OF CONTENTS

ACCEPTANCE SHEET	ii
ABSTRACT	iii
ACKNOWLEDGMENTS.....	iv
TABLE OF CONTENTS.....	v
LIST OF FIGURES	x
1. INTRODUCTION.....	1
1.1 Background.....	1
1.2 Areas of Study.....	4
1.3 Current Developments	5
1.3.1 Theoretical and Practical Developments in Thematic Map Design	5
1.3.2 Application of Expert Systems in Cartography and GIS.....	8
1.4 The Objectives of the Project.....	11
1.4.1 Problem Identification.....	11
1.4.2 Objectives.....	11
1.5 Proposed Methodology.....	13
1.6 Presentation and Organization of the Thesis	15
2. KNOWLEDGE CODIFICATION.....	17
2.1 Introduction	17
2.2 Properties of Map Objects.....	17
2.2.2 Spatial Distribution of Geographic phenomena	18
2.2.2 Measurement Scale of Geographic Data.....	20
2.2.3 Relationships between Variables.....	21
2.3 Properties of Maps.....	23
2.3.1 Map Symbol Types.....	24
2.3.2 Visual Dimensions and their Characteristics.....	25

2.3.2.1 Visual Dimensions	25
2.3.2.2 Perception Properties	27
2.3.2.3 Combination of Visual Dimensions	28
2.4 Other Factors in Map Design.....	30
2.4.1 Purposes of the Map.....	30
2.4.2 Screen-Based Maps vs. Paper Maps	30
2.4.3 Capacity of the Map	31
2.4.4 Cartographic Conventions and Map Aesthetics	33
2.4.5 Map Contents and Multiple Variable Mapping	34
2.5 Cartographic Design vs. Geographic Data	35
2.6 Summary	35
3. ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS	36
3.1 Introduction.....	36
3.2 Historical Sketch and Application Review	36
3.3 Knowledge Representation.....	39
3.3.1 Logical Representation Schemes.....	40
3.3.2 Production Rules.....	43
3.3.3 Semantic Networks.....	43
3.3.4 Frames	44
3.4 Solution Search.....	44
3.4.1 State Space Search.....	45
3.4.2 Direction of Search – Forward Chaining and Backward Chaining	46
3.4.3 Ordering of States – Depth-First and Breadth First.....	47
3.4.4 Heuristic Search.....	48
3.5 Expert System Architecture	48
3.6 Languages for AI Problem Solving and Expert System Shells	51
3.7 Summary	53

4. THE CONCEPT DESIGN OF TMDAS.....	55
4.1 Introduction	55
4.2 User Requirement Analysis and Problem Identification	55
4.3 Approaches in Problem Solving.....	57
4.3.1 Theme Oriented Approach	57
4.3.2 Cartographic Function and Data Property Oriented Approach.....	59
4.4 System Functionality and Modeling.....	62
4.4.1 An Optimal Framework	63
4.4.2 The Modified Framework.....	64
4.4.2.1 Information Query Module.....	65
4.4.2.2 Map Type Selection Module	65
4.4.2.3 Map Type Combination Module.....	66
4.4.2.4 Visual Dimension Selection Module	68
4.4.2.5 Actual Visual Dimension Design Module.....	68
4.5 User Interface and Explanation	70
4.6 Summary	71
5. KNOWLEDGE ACQUISITION	72
5.1 Introduction	72
5.2 Knowledge Structure and State Space Representation.....	73
5.3 The Design of Knowledge Acquisition Tables.....	75
5.4 Knowledge on Data Properties	77
5.5 The Map Type Categories	80
5.5.1 Symbol-Driven vs. Data-Driven	81
5.5.2 A Hierarchical Architecture of Map Type Category.....	85
5.5.2.1 Mapping Point Features	86
5.5.2.2 Mapping Line Features	86

5.5.2.3 Mapping Discrete Features.....	88
5.5.2.4 Mapping Area Features with Partial Coverage.....	89
5.5.2.5 Mapping Area Features with Total Coverage.....	90
5.5.2.6 Mapping Volume Features.....	91
5.5.2.7 Mapping Aggregated Data.....	93
5.5.2.8 Summary of Map Type Categories.....	95
5.6 Possibilities of Map Type Combination.....	96
5.7 Use of Visual Dimensions.....	98
5.7.1 Mapping Nominal Data.....	99
5.7.2 Mapping Ordinal Data.....	100
5.7.3 Mapping Absolute Quantitative Data.....	101
5.7.4 Mapping Derived Quantitative Data.....	101
5.8 Visual Constants and Further Aspects in Design.....	102
5.9 Knowledge Extraction from NAIS Data Set.....	103
5.10 Meta-Knowledge Extraction.....	104
5.11 Knowledge Validation.....	105
5.12 Summary.....	106
6. SYSTEM IMPLEMENTATION.....	108
6.1 Introduction.....	108
6.2 Environment for System Development.....	108
6.2.1 Selection of Platform.....	109
6.2.2 Selection of Programming Language.....	110
6.2.2.1 Turbo Prolog.....	111
6.2.2.2 The C Programming Language.....	115
6.3 The Design of System Performance.....	116
6.4 The Development of J.shell – an Expert System Shell.....	118
6.4.1 The Basic Format Used in J.shell.....	118
6.4.2 The Inference Engine of J.shell.....	124

6.5 TMDAS Knowledge Base.....	126
6.6 Explanation Facility in The Inference Engine	143
6.7 On-line Map Library	147
6.8 Summary	147
7. CASE TESTS AND ANALYSIS	148
7.1 Introduction	148
7.2 Cases and analysis	149
7.3 Summary	165
8. CONCLUSION	167
8.1 Summary and Conclusions.....	167
8.2 Suggestions for Further Development	172
BIBLIOGRAPHY	175
APPENDIX.....	185

LIST OF FIGURES

Figure 2.1	Three Maps Showing Different Relationships	22
Figure 2.2	Perception Properties of Visual Dimensions	28
Figure 3.1	AI Conceptual Overview.....	37
Figure 3.2	Laws of Equivalence.....	41
Figure 3.3	Laws of Equivalence for Quantifiers.....	42
Figure 3.4	An Example of Semantic Networks.....	44
Figure 3.5	An Example of Frames.....	45
Figure 3.6	An Example of State Space Representation.....	46
Figure 3.7	Depth-First vs. Breadth-First Search.....	48
Figure 3.8	Expert System Architecture.....	50
Figure 4.1	Theme-Oriented Approach.....	60
Figure 4.2	Cartographic Function and Data Property Oriented Approach	60
Figure 4.3.a	Optimal Framework for Automated Map Design	64
Figure 4.3.b	Modified Framework for Automated Map Design.....	64
Figure 4.4	Two Point Maps.....	69
Figure 5.1	Fundamental Differences between Human and Formal Knowledge.....	72
Figure 5.2	TMDAS Knowledge Structure.....	74
Figure 5.3	An Example of State Space Representation of Symbol Design	75
Figure 5.4	State Space of Map Design	81
Figure 5.5	Three "Point Symbol" Maps	82
Figure 5.6	A Hierarchical Architecture of Map Type Categories.....	85
Figure 5.7	Example of a Point Map	86
Figure 5.8	Example of a Line Map	87
Figure 5.9	Example of a Flow Map	88
Figure 5.10	Example of a Dot Map	89
Figure 5.11	Example of an Area Map.....	90
Figure 5.12	Example of a Category Map.....	91

Figure 5.13	Example of an Isoline Map	92
Figure 5.14	Example of a Located Graphics Map.....	93
Figure 5.15	Example of a Choropleth Map	94
Figure 5.16	Example of an Area Based Point Map.....	95
Figure 5.17	An Example of the Organization Level of Visual Variables.....	98
Figure 5.18	Visual Variables for Mapping Nominal Data	99
Figure 5.19	Visual Variables for Mapping Ordinal Data.....	100
Figure 5.20	Graduated Sizes in a Regular Pattern.....	102
Figure 6.1	Exploratory Programming.....	109
Figure 6.2	System Performance Design	116
Figure 6.3	An Example of a Recommendation from TMDAS.....	117
Figure 6.4	The Basic Control Flow of the J.shell Inference Engine.....	125
Figure 6.5	Reasoning in the Information Query Module.....	128
Figure 6.6	Example of the Data Relationships Represented by a Semantic Network.....	131
Figure 6.7	Reasoning in the Map Type Selection.....	136
Figure 6.8	Reasoning in the Map Type Combination	138
Figure 6.9	Reasoning in the Visual Dimension Selection.....	141
Figure 6.10	The Inference Tree Based on the Example Knowledge Base.....	146

1. INTRODUCTION

1.1 Background

Geographic Information Systems (GIS), an inter-disciplinary technology comprising geography, cartography, computer science, and a variety of other application domains, have been playing an important role in social and economic development through their use as decision making tools. Maps, either printed on paper or displayed on a screen, have become one of the principal outputs of GISs and been widely used as both a basis for decision making and a form of decision representation. There is a social need to produce high quality maps quickly. This means that cartography should be one of the main components of GIS. However, cartography continues to be the GIS component to which the least attention is paid. Buttenfield and Mackaness pointed out: "Although the principles of designing displays of spatial data have been investigated for centuries, very little use has been made of such principles in GIS... GIS needs to pay much more attention to visualization research and to the principles of good map design" (Buttenfield and Mackaness, 1991: 427).

Cartography, an old discipline, has been undergoing a revolution. The rapidly developing new technologies have made an unprecedented impact on cartography. This impact can be viewed from two perspectives: first, it shows the need for theoretical and conceptual development in the discipline of cartography; second, it offers cartography a chance to broaden its research area with the aid of new technologies. The use of new technologies depends on theoretical and conceptual development, which will provide a foundation and a rationale for further practical research. Computer Assisted Cartography (CAC) is one of the most active sub-areas of cartography in the use of new technologies. Many computing or drafting related cartographic functions can now be carried out with computers. A

number of powerful commercial software packages for PC-based thematic mapping are available such as GIMMS, ATLAS, PAMAP, etc.

Some of this software, developed from GIS and CAC, have led to the rapid development of another field, Electronic Mapping Systems (EMS). As a system for the development and/or the use of electronic maps (Siekierska and Taylor, 1991), EMS shares the same principles of GIS but emphasizes mapping and data display. Rich cartographic functions as well as a user friendly interface are normally available in these systems.

Well-known examples of such systems include the Electronic Atlas of Canada (EA) and the National Atlas Information System (NAIS). These are two related projects in the Canada Center for Mapping (CCM) of Energy, Mines, and Resources (EMR) Canada. NAIS, supported by the ARC/INFO package on SUN workstations, "is becoming a digital information service providing information at national scales, in particular, the 1:7.5 million scale. ... The NAIS 1:7.5 million database will be, within NAIS, the unique and centralized source of 'official' geographical digital data of Canada at that scale" (Cournoyer, 1991: 2). The EA, developed in house on SUN workstations, is designed to use data from the ARC/INFO data base and "... incorporates a wide range of cartographic functions needed for the creation and design of comprehensive electronic maps and for the performance of interactive cartographic analysis" (Siekierska and Taylor, 1991: 11). NAIS and the EA can offer users abundant information, and cartographic functions. However, cartographic knowledge dealing with the representation of the data in a graphic format must be provided by the user.

With the support of such new technologies, map making is becoming simpler and easier. Producing maps quickly no longer seems a problem. However, poor quality map products can often be observed (Mueller and Wang, 1990), because more and more non-cartographically trained users can access GIS, EMS or CAC related software and easily create maps. What is normally absent in these new mapping systems, for both the user and

the software, is cartographic knowledge. The cartographic component in the software is function-rich and knowledge-absent; this is becoming a serious problem that has raised the interest of the professional cartographic community. This absence of knowledge is due to the fact that unlike computing or drafting related cartographic functions which can be developed by arithmetic algorithms, knowledge is difficult to incorporate in conventional systems.

Expert Systems (ES) are often used to add such knowledge. As a branch of Artificial Intelligence (AI), ES is a technology efficient in manipulating symbolic knowledge and heuristics to simulate human experts solving real-world problems. Because of this distinctive characteristic, expert systems are considered to have great potential in cartography, especially in map design. Actually, expert systems in cartography are becoming an active research domain and receiving growing attention.

Buttenfield and Mackaness assert:

"The reliability of decisions based on poorly designed graphic products is not yet recognized as a problem by users of GIS packages...Cartographers should assert their knowledge of visual perception to improve the quality of visualized information in GIS: the need for reliable graphics is real, even if the demand is not yet recognized."

(Buttenfield and Mackaness, 1991: 427).

It was the recognition of this need that prompted the research project for the development of TMDAS – the Thematic Map Design Advisory System for geographic information systems and electronic mapping systems. Extracting the inherent knowledge of thematic map design, and inserting this knowledge into the automated mapping environment to safeguard against the creation of misleading maps, are the twofold aims of this project: expert system technology is used to accomplish these aims.

1.2 Areas of Study

The development of TMDAS is an interdisciplinary research project. The areas of concern include: cartography, geography, geographic information systems, electronic mapping, visualization, and expert systems. The following describe the specific aspects of each of these domains which apply to TMDAS.

Cartography: Cartographic design has been described as the most fundamental, challenging, and creative aspect of the cartographic process (Dent, 1990: 23). "It involves assembling, processing, and generalizing diverse data then symbolically displaying them as a meaningful, functional portrayal" (Robinson, et al., 1984: 139). Cartographic design consists of both the intellectual and the visual aspects which are sometimes in conflict, so that most choices have to be compromises. It is this cartographic knowledge that forms the kernel of the TMDAS knowledge base.

Geography: Mapping is a process used for communicating geographic information. Maps are graphic representations of geographic phenomena. Thematic maps are used to portray a vast range of themes in both physical and human geography. Thus, to understand the characteristics of different geographic phenomena, the spatial distribution of the phenomena, the measurement scale of the data, and the relationships among different features to be mapped is very important, because these aspects are the foundation upon which the cartographic representation should be based.

Geographic Information Systems and Electronic Mapping Systems: The aim of this project is to insert cartographic knowledge into the automated mapping environment provided mostly by GIS or EMS. An understanding of the principles and available cartographic functions of the GIS and EMS are thus necessary.

Visualization: This is a new research domain based on computer graphics. Visualization, the technique of using the power of direct visualization tools to stimulate visual thinking, has been studied by a variety of disciplines. Hence, there are different definitions of visualization arising from different perspectives. One of them, called geographic visualization, is defined by MacEachren (1992) as: "the use of concrete visual representations – whether on paper or through computer displays or other media – to make spatial contexts and problems visible, so as to engage the most powerful human information-processing abilities, those associated with vision" (MacEachren et al., 1992: 101). This definition can also be used as a definition of cartographic visualization. The research on visualization and associated new concepts and technologies provides more challenges and possibilities for map design. For instance, can the semiology based on conventional, printed graphics be equally applied to those produced in the electronic medium? Since the electronic medium is relatively new to cartography, much research is needed on this aspect.

Expert systems: The thematic map design advisory system aims to simulate human cartographers solving design problems. Expert systems, the technology efficient in manipulating symbolic knowledge and heuristics, is the technology to be used for the development of TMDAS: that of expert system. Therefore, a thorough understanding of the technology is the basis of the project.

1.3 Current Developments

1.3.1 Theoretical and Practical Developments in Thematic Map Design

Map design involves various sciences, such as psychology, physics, mathematics, statistics, computing science, etc. (Worth, 1989). "Graphic design is a vital part of cartography because effective communication requires that the various signs (lines, tones, colours, lettering, and so on) be carefully modulated and fitted together. ... the

cartographer – a map designer – must pay attention to the basic principles of graphic communication" (Robinson, et al., 1984: 137). A good map design depends on the understanding of both the properties of the geographic data to be represented and the procedure by which the map reader extracts information from a map. Therefore, both cartographic cognition and communication are important issues that must be considered in map design.

The theories of graphics representation presented by Bertin in his book "Semiology of Graphics" (Bertin 1983) are considered as the most comprehensive (Laurema, et al., 1991). The rules of the graphic system and the use of the graphic sign-system are discussed in detail in this work. Many of these rules have been applied in cartographic design. Some basic elements of map design are discussed in the book "Elements of Cartography" by Robinson, et al. (1984), which has been considered as one of the authoritative text books in cartography. In addition, as Buttenfield and Mark mention, "Recently published textbooks by Cuff and Mattson (1982) and by Dent (1985) are key sources of information on map design principles" (Buttenfield and Mark, 1991: 135). In these two books, named "Thematic Maps, Their Design and Production" (by Cuff and Mattson) and "Cartography, thematic map design" (by Dent 1985, 1990, the 2nd edition), thematic maps, map symbols and symbol dimensions are analyzed in detail, in relation with the nature of geographical phenomena and data. "Understanding maps" by Keates (1982) is another book dealing with maps and visual perception. The central theme of this book is to identify the principal factors in visual perception which affect both the design and the use of maps. Graphic representation methods and their classifications in thematic map design are discussed in detail in the book of Design of The Thematic Map (Zhang and Huang, 1980, in Chinese).

Most of the theories dealing with the representation of graphics in cartography are based on the making of conventional paper maps. Although several textbooks now deal with

computer assisted cartography, none appear to direct much attention to the design of maps for computer display (Gooding and Forrest, 1991). Recent interest in scientific visualization has emphasized the development of new and more powerful visualization tools such as computer animation and three dimensional modeling (MacEachren et al., 1992). But most of the research deals with the development of new technologies, rather than the methods of applying it. Slocum and Egbert (1991) reviewed current cartographic data displays and grouped them into three types: static mapping, interactive mapping and animated mapping. Some of the techniques used in these mapping procedures are quite different from those of conventional mapping. An example is the sequencing concept developed by Taylor (1987) for interactive mapping. In sequencing, classes are displayed in a specific order so as to enhance the interpretation by providing the reader with "chunks" of information (Taylor, 1987) and by emphasizing the ordering of the data (Slocum and Egbert, 1991). Blinking and movement are other powerful visual stimulators which can be used in the electronic medium. An experiment which examined different perceptions between the interpretation of screen based maps and printed maps was carried out by Gooding and Forrest (1991). In their experiment, the screen based maps were video pictures of existing printed maps. The result proves that interpretation of this kind of screen based map is not as good as that of paper maps. Some general observations, such as that "the contrast of the screen image appears to be less than on the printed maps" (Gooding and Forrest, 1991: 18), as well as specific problems, such as that certain elements were more difficult to be found on the screen based map than on the printed map, are described in the paper. The authors concluded that screen based maps should not be simple copies of paper maps. Actually, many possibilities and limitations of screen based maps require investigation, and intensive experiments and studies based on screen graphics, cognitive, and psychological perspectives are needed before systematic and significant conclusions can be drawn.

In general, two points need to be emphasized. First, there are some basic theories and principles which can be used in expert system development in thematic map design, although they may exist in formats which are very different from the one required by formal knowledge representation. These theories and principles form the basis of the knowledge body. On the other hand, it should also be realized that this body of knowledge is not as complete as expert system development may require. Some research effort is needed. The first point provides the feasibility of the project, while the second point adds challenge to the project.

1.3.2 Application of Expert Systems in Cartography and GIS

Expert systems technology applied to cartography can yield significant benefits in the form of increased productivity and increased product uniformity (Bossler, et. al., 1988). In cooperation with computer assisted cartography, a number of research projects on cartographic expert systems have been carried out and some prototypes have been developed. The research dealing with expert systems in cartography can be divided into four classes: map projection selection, such as MAPKBS developed by Nyerges and Jankowski (1989); geographic name placement, such as AUTONAP developed by Freeman and Ahn (1984); map generalization, such as MAPEX developed by Nickerson and Freeman (1986); and, map design, such as ESSYD developed by Wang (1992). Map design is one of the most active research areas.

Map design is characterized as a good candidate problem for expert systems except for the fact that the task does not have a narrow focus (Laurema, et al., 1991 and Wang, 1992). However, since map design as a whole can be decomposed into sub-tasks, the specific system development of the sub-tasks can be performed, and an incremental development strategy can be adopted. This is the strategy applied to most map design prototype systems.

The "knowledge-based query system for cartographic visualization", developed by Laurema, et al. (1991) is a prototype system specially focused on land-use mapping. Nineteen specific map types have been defined, such as constant point (equal-size) map and variable point (unequal-size) map. Specific knowledge on both the land-use and an additional thirteen socio-economic attributes associated with municipalities is stored in the knowledge base. Users can choose topics from a list of themes, and possible map types are recommended. The NEXPERT an expert system shell which has the ability to support both a reasoning process and an object-oriented representation, is employed in the prototype.

ESSYD, an expert system for symbol design, developed by Wang (1992), is another system implemented using Prolog language. Conceptual graphs are used in some of the knowledge representation in this system. Fourteen specific map types are defined according to the number of variables (maximum of 2, except for multiple absolute information which can be shown by compound symbols), the level of measurement scale of the data, and the dimensions of the symbols such as single nominal point map, or double ordinal area map (Wang, 1992). Selection of the visual variable specification is also developed in this system, especially for colours.

There are other systems, such as CES (Cartographic Expert System) by Muller et al. (1986), MAP-AID by Robinson G. and Jackson (1985), ACES (A Cartographic Expert System) by Pfefferkorn, et al. (1985), MAPKEY, "a knowledge base system based on GIS for thematic mapping" by Zhang, et al. (1993), and PC-mapper, by Hua and Gao (1993). They focus on different aspects of cartographic design. A review of some of the expert systems in map design mentioned above will be given in the later chapters.

The integration of artificial intelligence and GIS is a trend in GIS development. Much research has been carried out in this area. Smith and Yiang (1991) discussed that the most important approaches to the application of knowledge-based techniques (KBT) in GIS can

be characterized into five groups: acquisition, storage, access, analysis and processing, and interface. Some of the systems are: KBGIS-II, a knowledge-based geographic information system by Smith, et al.(1987), PROSTGRES, a rule manager system by Stonebraker and Hansen (1988), ASPENEX, an expert system providing interface services and interfacing to the MOSS GIS by Morse (1987), and LOBSTER, a query language for GIS, serving as an interface to an object-oriented network spatial DBMS by Frank (1988).

A review by Robinson V. B and Frank (1987) on the use of expert systems within the general field of GIS for about 20 systems pointed out four areas where expert systems may prove useful in the development of a GIS; these are: (i) map design (ii) terrain feature extraction (iii) geographic database management, and (iv) geographic decision support. The last area, geographic decision support, is now often called spatial decision support systems (SDSS) (Waters, 1991). Robinson V. B and Frank (1987) also suggested that map design, land-use classification, and physical feature extraction were likely to prove both rewarding and challenging problems in the immediate future. The survey also showed that the current expert systems were loosely coupled to the data handling capabilities of GIS, because of the different inherent characteristics between expert systems and GIS (Smith and Yiang, 1991). High level integration of expert systems and GIS is still an open problem.

In summary, with the development of theoretical and conceptual research in cartography, there is a definite body of advice and knowledge which can be used to find effective solutions to the problems of map representation, although there are no universally accepted rules (Muller, et al., 1986). Cartographic design processing which mixes theories, and experience, is a suitable problem for expert systems. Certain progress has been made, but much more effort is needed towards the development of an operational system, especially in concept design and cartographic knowledge acquisition and representation aspects.

1.4 The Objectives of the Project

1.4.1 Problem Identification

Thematic map design involves decision making on the following elements: scale selection, projection selection, base map design, geographic data processing, graphic representation, map layout, etc. Each of these elements is a complex procedure for automation. For any given data set, a good map designer should seek "an organized whole for the graphic elements on the map to achieve an efficient and accurate transfer of knowledge between map author and map user" (Dent, 1990: 24).

TMDAS focuses on the graphic representation aspect. For a given data set, the designed system should be able to find: (i) whether the variables in the data set should be combined into one map; (ii) if so, whether and how they should be grouped into one map; and, (iii) what kinds of symbols and visual variables should be used for each variable. Specific knowledge extraction will be carried out on NAIS data set to make the use of the NAIS data more convenient, while the system should work for any generic mapping requirement using any data set.

1.4.2 Objectives

The objectives of this project are to extract knowledge in thematic map design, and insert this knowledge into an automated mapping environment through the use of expert system technology. This is a knowledge engineering project, which consists of the following aspects:

Concept Design: The purpose of this project is to insert the knowledge of map design into an automated mapping environment, such as GIS or EMS, by using expert system technology. The development of such an operational system should be considered a long term engineering project. Therefore, the concept design of the system is very important.

The framework of a concept design will be given based on the analysis of user requirements, procedures involved in map design, and possible problem decomposition. A clearly defined concept of modules and the logical inter-relationships among the modules will be presented.

Knowledge acquisition: The acquisition of the knowledge involved in thematic map design is the core of this project. Knowledge will be mainly extracted from text books, maps, and atlases. The knowledge will also be codified in a format which is easy for expert system implementation. The specific knowledge on the NAIS data set will be extracted from the National Atlas of Canada – Fifth Edition. A series of knowledge acquisition tables will be constructed.

Knowledge representation: Knowledge representation is a very important factor for reasoning inference and hence will influence the efficiency of the system. Since the knowledge involved in thematic map design is quite intensive, knowledge modelling and knowledge representation schemes for different types of knowledge have to be carefully studied to reach an optimal approach.

System implementation: A prototype will be developed based on a proper platform, and a software environment investigated for the system. Prototyping has always been considered an important aspect of expert system development. In this project, it will be used to test the proposed framework, validate and modify the knowledge base, examine the efficiency of the knowledge representation scheme, and determine the feasibility of the project for further development.

Case tests : The system must be tested by using real cases to compare the recommendations given by the system with those used by human cartographers. This must be accomplished in order to debug the system, enhance the knowledge base, test the user interface, and draw conclusions for improvement.

1.5 Proposed Methodology

The development of TMDAS is a knowledge engineering project, which is quite different from conventional system development. Experience from successful expert system development projects can be found in the literature and must be considered carefully. General procedures applied to knowledge engineering have to be followed, with consideration of the specific problems in thematic map design. The following are the methodologies to be applied.

Problem definition and concept design: This is the first stage and is very important in expert system development. It is very helpful to identify the issues, the requirements, and the limits of the research domain as clearly as possible at the beginning. It is also important to clearly describe what the system can and can not do. The system design must be open-ended to allow for further development.

Knowledge codification: After the problem is clearly identified, the codification of knowledge will proceed in order to identify the kinds of knowledge are needed, the manner in which they exist in real life, the extraction methods, the possibilities of decomposing, and the subtasks and inter-relationships among them. This procedure will produce a solid foundation for concept design and knowledge acquisition.

Knowledge acquisition: Knowledge acquisition is considered a key procedure in expert systems development, in which knowledge has to be described in an explicit format for system development. For this project, knowledge has to be extracted mainly from the literature: published books, papers, maps and atlases, but may also be obtained from experienced cartographers. In particular, the knowledge extracted from the National Atlas of Canada – Fifth Edition will form part of the knowledge base in the system.

Knowledge representation and system design: "The choice of a scheme for the representation of knowledge is critical to successful problem solving. It can affect the number of stages in the problem domain and thus can either increase or decrease the complexity of the problem" (Schutzer, 1987: 14). This stage of the project involves both the representation of the knowledge in the system's accepted format and the actual storage of the knowledge in the system. Different representational methods must be reviewed and the system design will be based on an optimal knowledge representation scheme.

Prototype development: Unlike in conventional systems, a good knowledge representation scheme remains unknown until it is achieved in expert system development. As Schutzer pointed out: "Usually, however, no clear, nature means of representing the knowledge associated with a given problem emerges" (Schutzer, 1987: 15). The development of a prototype is, therefore, essential.

Case tests and analysis: Knowledge validation is an important issue in expert system development, because unlike mathematics or statistics where the rules may be accepted as universal, map design can have a considerable number of possible solutions for any one problem. In addition, human knowledge can be mis-represented by formal knowledge. Therefore, the case test is an efficient method to verify if the knowledge is valid, if the system fulfills the requirements, if the system is efficient, etc.

The stages in expert system development are less well-defined than in ordinary computer systems development and they often overlap a great deal (Hart, 1986). Awareness that it might be necessary to review objectives at every stage of development if problems become more complex than was originally anticipated is desirable, and a workplan, which incorporates this consideration, must be prepared.

1.6 Presentation and Organization of the Thesis

As a research project, there are two components presented: the prototype system of TMDAS, and this thesis.

The prototype of TMDAS was developed using Prolog and the C programming languages. The system occupies about 2MB of disk space, and can be run on any IBM compatible computer with a VGA graphic card. There is on-line help available to the user. The system consists of two parts: one is named J shell, and is a generic expert system shell with an inference engine and explanation facilities. It can be used for any other knowledge base developed using the J.shell format. The other is the TMDAS knowledge base, which is constructed following the concept design of the system and contains the knowledge acquired in the project. It has about 500 knowledge entries; they are divided into 6 modules because of the run time memory limitation. Windows, menus, as well as an on-line map library are designed to improve the user interface. Preliminary testing has provided positive feedback on the system.

This thesis describes the research for this project. It consists of eight chapters. Chapter 1 is the introduction to the project, and it explains the reasons why this project has been undertaken, the purpose of the project, the objectives to be realized, and the methods for achieving these objectives. The next chapter deals with knowledge codification for map design, based on some authoritative English text books. The knowledge codified in this chapter provides the basis for both the concept design and the TMDAS knowledge base. Chapter 3 is an introduction to the technology used in the project; the basic principles as well as some application issues in the development of expert systems are discussed here. The following chapter concentrates on the concept design of an operational expert system in thematic map design. After a discussion of different approaches, the framework used in TMDAS is presented. Chapter 5 deals with knowledge acquisition, and the procedure for

transforming human knowledge into formal knowledge is discussed. In addition, some further study and development of the knowledge is presented in order to make a solid knowledge body available for TMDAS implementation. Chapter 6 presents the prototype implementation of TMDAS. The technical issues are discussed module by module with some examples and pseudo codes. Chapter 7 discusses case tests and analysis. Eight cases selected from the preliminary test are presented. The cases are selected from existing atlases, so that the recommendations given by the system can be compared with the designs by human experts. This is followed by the last chapter, which presents the conclusions and discusses the further development of TMDAS. The actual system running procedures for the eight cases were recorded and are included in Appendix.

2. KNOWLEDGE CODIFICATION

2.1 Introduction

The development of TMDAS involves knowledge dealing with several domains such as: geography, cartography, graphic semiology and computer graphics. In this chapter, an overview of the knowledge domain will be presented and the knowledge will be codified from expert system development perspectives. A knowledge frame for the development of TMDAS will be presented in this chapter, although it may be found that there is less knowledge available in some parts of the frame. The knowledge required for these parts will be further discussed in chapter 5. The knowledge codification carried out in this chapter is based on several authoritative English text books (such as Robinson, et al., 1984, Dent, 1990, Cuff and Mattson, 1982, and Bertin, 1983).

2.2 Properties of Map Objects

Maps are considered as vehicles for the transmission of information. "In general, all maps may be classified as either general-purpose or thematic types" (Dent, 1991: 7). General-purpose maps emphasize location. An example of this type of map is the topographic map. Thematic maps aim to illustrate the structural characteristics of some particular geographical distribution (Robinson, et al., 1984). The International Cartographic Association defines the thematic map as "A map designed to demonstrate particular features or concepts. In conventional use this term excludes topographic maps" (Dent, 1991: 8). The thematic map design discussed in this thesis is based on this type of map.

Thematic mapping involves the mapping of natural and/or cultural phenomena or of the abstract ideas about them (Dent, 1991). There is no limit to the subject matter of thematic maps, "and they range in appearance from the satellite cloud cover image to the shade map of election results" (Robinson, et al., 1984: 9). It is clear that no cartographer can be an

expert in all the subject domains. Actually, from the graphic design perspective, "cartography is independent of subject matter" (Robinson, et al., 1984: 11). But, on the other hand, to achieve a cartographic design, "the cartographer must be *expert in analyzing the data* that need to be mapped" (Cuff and Mattson, 1982, emphasis added by the author). Based on these two aspects, it can be concluded that, from a cartographic perspective, there must be generic data properties to all map objects, which can be used as control factors in map design. These generic data properties are the spatial distribution of the geographic phenomena, and the measurement scale of geographic data (Robinson, et al., 1984; Dent, 1990).

2.2.1 Spatial Distribution of Geographic Phenomena

It is necessary to make a distinction between the two terms of geographic phenomena and geographic data. *Geographic phenomena* can be defined as elements of reality that have spatial attributes, while *geographic data* are the facts chosen to describe geographical phenomena (Dent, 1990: 112). For example, a *city* is a geographic phenomenon, while the *total population of a city* is data to describe the size of the population of the city. Spatial distribution is related to geographic phenomena, and measurement scale is related to geographic data.

Spatial distribution deals with the pattern exhibited by the geographic phenomena. It consists of two components: continuity and spatial dimension.

Continuity: There are two types of phenomena with respect to continuity: discrete and continuous. *Discrete geographic phenomena* refers to phenomena "that do not occur between spatial observations, that form a separate entity, and that are individually distinguishable from place to place" (Dent, 1990: 96). *Continuous geographic phenomena* refer to phenomena which exist spatially between observations (Dent, 1990).

Spatial dimension: Spatial dimension is here defined as the form of geographic phenomena in the generalized cartographic model of the world (rather than in the natural world itself), while the generalization of the model depends on the scale of the map. Dent points out that "an interesting feature of phenomena is that their form is intricately related to scale and may change with the level of inquiry" (Dent, 1990: 95). For example, a city can be an area feature on one map, and a point feature on another. It is difficult to define point features or line features in the natural world; they become points or lines only when the scale is so small that they can not be represented as areas any more. Since this definition is based on the generalized model which is already related to the map scale, as long as the map scale is decided, the spatial dimension of the mapping phenomena is defined. The selection of map scale is not included in the project.

Mapping is a procedure of transferring reality in the three dimensional real world into a representation in a two dimensional limited space. It is a very complex procedure. This generalized cartographic model can be considered as a conceptual reduction of the mapping world in three dimensions according to the map scale. In this model, the following types of spatial dimensions of geographic phenomena can be defined as (Dent, 1990):

Point: zero-dimensional

Line: one-dimensional

Area: two-dimensional

Volume: three-dimensional

Space-time: four-dimensional.

Spatial distribution of geographic phenomena plays an important role in the design of a map created to show the true distributions of the map objects. There is another type of map, however, which shows the data by aggregating it in regions. In this type of map, the key issue is the selection of *geographical area units* by which the statistical data is aggregated. For instance, the map object can be the number of post offices in each county (point feature), the total cultivated area in each region (area feature), or the length of highways in

each province (line feature). Census units or administration divisions are often used to tabulate aggregated data for this type of mapping.

In general, the spatial distribution of geographic phenomena is one of the main factors to be considered in map design. It affects the type of map to be made, and the kind of symbol to be used.

2.2.2 Measurement Scale of Geographic Data

Measurement scale is a scaling system to structure observations about reality. A widely used scale system is a four level precision system defined as following (Dent, 1990; Robinson, et al., 1984):

Nominal scale: objects are grouped according to a certain criteria in which only equality or inequality comparison can be carried out, no arithmetic operations can be performed between the groups.

Ordinal scale: objects are ranked according to certain non-numerical order from least to most or vice versa so that the information obtainable is of the "greater than" or "less than" variety and no more quantitative data can be inferred.

Interval scale: objects are assigned to enumeration data and can be arrayed in order of rank, and the distance between ranks are known. But there is no natural origin in this magnitude scale, and the values are relative.

Ratio scale: like interval scale, objects are assigned to enumeration data and can be ranked, and the distance between ranks are known. The difference is that there is a natural origin in this scale, and the values are absolute.

This scale system is widely applied in geography, as well as in cartographic data processing. It is, however, not suitable for cartographic symbol design. "From the point of view of map making there is no difference between the symbolization of geographic data categorized in interval and ratio scale" (Robinson, et al., 1984: 110). It is not necessary to distinguish these two for symbol design. There is, however, an important aspect in enumeration data which does make a significant difference, that is, whether the data are absolute quantitative or derived quantitative. For application convenience, therefore, the interval and ratio scales will be replaced by the following two scales to form cartographically applicable measurement scales:

Absolute quantitative: objects are assigned to enumeration data which has *total* properties. The distance between groups can be derived, and the data can be either interval or ratio data.

Derived quantitative: objects are assigned to enumeration data which has *rate* properties and are derived values. This kind of data can be ratios involving area or ratios independent of area.

Examples of absolute quantitative data include total population and number of cars, while examples of derived quantitative are population density and sex ratio. These two kinds of data usually involve completely different map designs. Measurement scale is the key factor in the selection of visual variables in map design.

2.2.3 Relationships between Variables

Variable is the term used, in this thesis, to refer to the attribute stored in the database and associated with locational data. In thematic map design, more than one variable is often represented in a map sheet by using various symbols. The maps A, B, and C shown in figure 2.1 are examples. As stated on the right of the maps in the figure, each of the maps contains three variables, but the graphics used in each map for combining the three

Variable 1: Light industry
Variable 2: Heavy industry
Variable 3: Total output

Variable 1: Population in Cities
Variable 2: Dominant language
Variable 3: Transportation

Variable 1: Pump stations
Variable 2: Oil pipelines
Variable 3: Oil associated fields

Fig. 2.1 Three Maps Showing Different Relationships

variables are very different. The reason is that the inherent inter-relationships between variables in each group are different. Relationships between variables, therefore, play an important role in multiple variable map design. They specify the reasons for putting variables together and the methods for doing so. Human cartographers often need to have some knowledge of the data set, and, based on their experience, they decide which graphic schemes to use. In the automated mapping environment, however, the computer programs may know no more than that the maps are all three variable maps, and they cannot distinguish between them if there is no knowledge represented explicitly. To produce the three different designs as a human cartographer does, shown in figure 2.1, in the automatic mapping environment, therefore, knowledge of the relationship between variables is needed. In conventional mapping, however, this procedure seems to be carried out by cartographers naturally, based on their knowledge, and it has not been discussed in the literature. In this project, some research to establish generic criteria for classifying the relationships between variables has been carried out. The definitions of different types of relationships and the mechanisms applied in this project will be introduced in chapter 5.

2.3 Properties of Maps

Maps have always been considered products of the combination of science and arts. A high quality map should be: 1) scientifically correct, which means that the map has to transfer information accurately to the map reader, and, 2) artistically enjoyable, which means that the map should also be attractive to the map reader. These two aspects are considered as two goals for cartographers. The scope of this project is mainly focused on the first goal, namely accuracy.

Even when the focus is narrowed to the first level, maps remain complex objects. Simply picking up two maps from a map library, one can find the difference between them. Each map is unique. In this aspect, maps share the same characteristics as map objects: the number of different map designs, just like the number of different objects to be mapped, is

infinite. Therefore, we must also (just like what we did in analyzing mapping objects) determine generic *map properties* which can be used as control factors in map design. These properties are map symbol types and visual dimensions.

2.3.1 Map Symbol Types

"There is obviously an unlimited variety of spatial data that can be mapped, all of which must be represented by symbols. In order to consider the ways in which signs can be employed to symbolize the variety of data, it is helpful to classify them" (Robinson, et al., 1984: 140). One method of classification is by *symbol types*, which consist of three generally recognized symbols defined as ¹ :

Point symbol: individual symbol which can be located by a pair of coordinates (0-dimension).

Line symbol: individual linear symbol which must be located by two or more pairs of coordinates (one-dimension).

Area symbol: symbols whose extension must be defined by an enclosed polygons (two-dimension).

It should be emphasized that although the words "point", "line", and "area" have been used in both cases of spatial dimension of geographic phenomena and map symbol types (the geometric dimension of the symbol), there is no definite match between these two. Robinson points out that "unfortunately not all place data are mapped with point symbols. If they were, the cartographer's symbolization problem would be easier. Likewise point symbols can map other than place data" (Robinson, et al., 1984: 286). Point symbols can also be used to represent absolute quantitative data aggregated at points (e.g. total cultivated land in counties, which is an area feature), and line symbols to represent volumetric data

¹ Two other kinds of symbols, three-dimensional diagrams and shading may be added to this list, and are used to represent volumetric data such as the land surface. Because the construction of those symbols is quite complex and their usage is also very different from those of the more conventional symbols, they are not discussed in this project.

(e.g. isoline maps for temperature). Failure to make significant distinctions between these two concepts can result in a misleading map design. There is no direct link between the spatial dimension of the map object and the type of symbol used. The link, however, is important in the automated environment, and can be established indirectly. This will be discussed in detail in chapter 5.

2.3.2 Visual Dimensions and their Characteristics

The visual dimension is one of the key factors in map design. As data to be represented has certain properties, each visual dimension has its own inherent characteristics. To understand these characteristics and select the proper visual dimension for the specific data properties, is a fundamental process in making a correct map design.

2.3.2.1 Visual Dimensions

Visual dimensions refer to the primary graphic elements of symbols in graphic design. They are defined in such a way that a difference in graphic appearance can be achieved by changing one or more of these visual dimensions. Various authors use slightly different terminologies and classifications. The term "visual variables" used by Bertin (1983) has six components: size, value, texture, colour, orientation, and shape. "Primary graphic elements" is the term used by Robinson, et al. (1984) and they have seven elements: hue, value, size, shape, spacing orientation, and location. The term "symbol dimensions" is used by Dent (1991) and defined as: shape, size, colour hue, colour value, colour intensity, pattern orientation, pattern arrangement, and pattern texture.

Based on the analysis of these concepts, *visual dimensions*, as a term, is defined in this project: hue, orientation, form, value, spacing and size.

Hue: the attribute of colour related to wavelength. Colour is often used in practice to refer to hue, but since there are two other components of colour, value, and saturation, hue is considered as the more precise term.

Orientation: the directional arrangement of an elongated individual symbol.

Form: the graphic characteristic provided by the distinctive appearance of a regular configuration (such as a circle, or a square).

Value: the relative lightness or darkness of a mark, whether black or any other hue. It is a convention to associate "more" with a darker value and "less" with a lighter value for paper maps while this is reversed for screen-based maps.

Spacing: the distance between individual symbols which form a sign, such as a series of dots or lines.

Size: the diameter, area, width, height, etc., according to the dimension of the symbol.

In map symbol design, visual dimensions are actually used in two different ways, varied or uniform. In this research project, the distinction has been made according to the way in which it is used. One type of visual dimension will be called visual variables and the other, visual constants. *Visual variables* are the visual dimensions which change in appearance when the data changes, while *visual constants* are the visual dimensions which do not change with variations in the data. For example, when a point symbol map is selected to map absolute quantitative data at points (e.g., population of cities), the size of the symbol should change as the data changes, while a specific form such as a circle, or a square is used to construct the symbol. Here, size is the visual variable and form is the constant.

2.3.2.2 Perception Properties

Data measurement scales, as discussed earlier, can be grouped into four levels for map design purposes: nominal, ordinal, absolute quantitative, and derived quantitative. In comparison, visual dimensions also each have their own organization level. Bertin stated that "... 'level' is probably the greatest potential source of graphic error. Level of organization assumes particular importance in cartography" (1983: 64). Bertin organized data measurement properties into three levels: qualitative, ordered, and quantitative, and named this *the level of organization of the components* (see Bertin, 1983: 34-9). He indicated that "Each visual variable has its particular properties of length and level. It is important that each component be transcribed by a variable having at least a corresponding length and level" (Bertin, 1983: 7).

The level of organization of visual dimensions is based on the perception properties which are the inherent characteristics of the visual dimensions that the viewer senses. There are four levels of perception properties described by Bertin (Bertin, 1983: 65-9):

Associate perception: the property of showing the similarity and difference among symbols;

Selective perception: the property of grouping and arranging symbols visually;

Ordered perception: the property of showing symbols in a specific sorted order;

Quantitative perception: the property of showing symbols in order and the difference can be quantified.

Tests of perception properties were carried out by Bertin (1983) and further developed by Bos (1984) for each visual variable referring to each level under a "pure" condition, that is, with all other variations excluded. Figure 2.2 is the summary of the relationship between visual variables and perception properties. Associative, ordered, and quantitative perceptions of the symbols are related to nominal, ordinal, and quantitative measurement scales of the data respectively, while selective perception affects the ability of the map reader to grasp overall information. The perception properties are based on a large number of tests and are widely accepted.

	Visual Variable					
Perception property	Hue	Form	Orientation	Spacing	Value	Size
associative	+	+	+	0	-	-
selective	++	-	0	+	+	+
ordered	-	-	-	0	++	+
quantitative	-	-	-	-	-	+

(++ = very good; + = good; 0 = moderate; - = poor/no)

Figure 2.2 Perception Properties of Visual Dimensions (adapted from Bos, 1984)

2.3.2.3 Combination of Visual Dimensions

What has been described so far is based on the use of a single visual dimension. In mapping practices, however, experienced cartographers often use more than one visual dimension. Bertin (1983) has studied the possible combinations of visual dimensions and stated that all the combinations of visual variables are possible. In Bertin's study, all

possible combinations of visual dimensions can be grouped into two categories: one is called redundant combinations, and the other is called "meaningful" combinations.

Redundant Combination

Redundant combinations are defined by Bertin as "a combination of several variables, utilized to represent only a single component" (Bertin, 1983: 187). This is used quite often by experienced cartographers, because "redundant combinations increase the separation between the steps of the retinal variables" (Bertin, 1983: 183). Bertin stated that "a combination of variables will retain the properties of the variable having the highest level of organization, ... (with the exception of the size-value combination)" (Bertin, 1983: 186). Since the use of a redundant combination involves a higher level expertise, it is beyond the main purpose of this project. However, a redundant combination commonly used in making choropleth maps, namely the combination of value and hue, is included in TMDAS. As Bertin described, this value-hue combination retains the properties of value, since value has a higher level of organization than hue. The steps between symbols using value-hue, and the length of the variable to be depicted are increased (for details on the concept of length, refer to Bertin, 1983). In fact, the combination of value and hue is often used to represent data in which qualitative change occurs as quantitative values change. In temperature maps, for example, a "warm" colour can be used for a temperature above 0°C, while a "cool" colour is used for temperatures below 0°C.

"Meaningful" Combination

If two visual variables are each associated with a different item to be represented, and form one integrated symbol, the combination is no longer redundant. Bertin called this type of combination a "*meaningful*" combination. It is difficult to predict the overall perception of different combinations; the design must be done carefully in order to satisfy the legibility of both variables. The key point is that the combination should really be meaningful, and

obviously the two items can not involve the same visual variable. There are quite a few maps designed using this technique in the Fifth Edition of the National Atlas of Canada, such as the map of Canada Sawmills, in which the variables of sawmill lumber production (in units of thousands of cubic meters) and wood type are represented using this "meaningful" combination point symbol. "Meaningful" combination employed in point symbols is included in TMDAS.

2.4 Other Factors in Map Design

2.4.1 Purposes of the Map

Unlike general-purpose maps, thematic maps are normally made for a specific anticipated audience and usage. The purpose for which the map is being made is, therefore, one of the factors in determining its final form. The scope of the specific objectives can affect the selection of contents and their graphic representations. For example, a map designed for schools should be much simpler than one for a scientific analysis, and the design of a tourism map should be different from a road map in the use of symbols. Knowledge in this area is also related to the study of map use. In this project, little research was done on this aspect. In the prototype of TMDAS, this factor has been simplified to one point: whether the map is to be a simple map (maximum 3 layers) or a complex map (maximum 5 layers); this can be replaced by a complex module later on.

2.4.2 Screen-Based Maps vs. Paper Maps

The visual dimensions and their perception properties described above are all based on paper maps. The electronic medium can offer many more possibilities of map design. This has already become an interesting research domain in cartography. Multi-media and animation are some of the active cartographic research topics. However, this research is still in the beginning stages. Most of the work has emphasized the development of new techniques rather than the use of them. For example, in addition to symbol types and

visual dimensions, there may be other factors, such as sound, to be added as basic properties for electronic maps. Moreover, there can be other visual dimensions, such as blinking, which can also be used. Some experiments have been carried out and described in the literature, such as sequencing, blinking, and movement (for sequencing, see Taylor, 1987). But much more research and experiments are needed to study readers' perception of these new techniques. For example, it is necessary to know that under which conditions, these techniques can be applied. In the case of sequencing, does it imply an ordinal perception? If movement is considered in the design, what would its effect be on the legibility of the map? Could the density of the map be increased or decreased when blinking is used? It is still very difficult to draw any conclusions on these aspects at this time. Intensive experiments and systematic research are needed to investigate new possibilities of graphic representation and the cartographic characteristics of these new methods.

It is the intention of this project to take some new representation methods into consideration for the development of TMDAS, but the primary objective of the study is to codify available knowledge in map design and represent it in the form of machine intelligence rather than to study new representation methods. Although both paper maps and screen-based maps have been included in the system, the differences in making these two kinds of maps are only addressed in two aspects: the capacity of maps and the use of conventional visual dimensions.

2.4.3 Capacity of the Map

The capacity of the map is related to the density of the map, but it is not determined by it. Other factors, such as clarity and legibility, visual contrast, visual balance, the figure-ground effect, and the hierarchical structure of the contents, also play important roles in the capacity of the map.

Clarity and *legibility* refer to the proper size of the symbols and the significant differences between them. They depend on the medium which carries the map, and on the probable use of the map. But, "the symbols of a map are large enough to be seen does not in itself provide clarity and legibility" (Robinson, et al., 1984: 146). *Visual contrast* is also important in order to distinguish individual symbols. Contrast needs to be carefully used to avoid the two extremes of too high or too low. *Visual balance* involves the positioning of all items on the map in such a way that their relationship appears logical and a balanced visual weight is kept. The *figure-ground* phenomena is a complex occurrence in human visual perception. It refers to the process of simultaneously separating a whole display into two parts: a figure on which the eye settles and which it sees clearly, and the amorphous, more or less formless background (Robinson, et al., 1984). Moreover, hierarchical structure is sometimes used to show more complex relations among variables, and requires a kind of internal graphic organization or structuring. All these factors affect each other. A good map design depends on a type of comprehensive analysis and requires that the designer work with the graphics in an interactive manner. This means that it would be very difficult to effectively utilize all of them in a fully automated environment. But the capacity of the map is such an unavoidable issue that in any map design expert system, it must be considered, even though not all the factors mentioned before can be taken into account. Actually, in most of the expert systems developed, the number of variables is often used as the control factor for map capacities (see Laurema, et al., 1991 and Wang, 1992). As it can be seen, this method is much simpler than the comprehensive factors mentioned before. To bring the approach closer to the cartographer's, more factors have been included in the method applied in TMDAS. In order to introduce this method, the concept of map layer will now be introduced.

A *map layer* is defined as the basic graphic unit which is associated with one locational data file related to one type of geographic phenomena. Although the locational data may be related to one or more variables (attribute data), the variables will be represented by one

symbol. The symbol can be a simple symbol, or a compound symbol, or may use a combination of visual dimensions. Therefore, the variables must first be grouped into layers according to their relationships, and a map layer is the basic unit for symbol design. Based on this definition, the maps in figure 2.1 each involve a different number of map layers, although they all contain three variables. Map A is a one layer map, map B is a two layer map, and map C is a three layer map. As can be seen from the examples, the number of map layers can be equal or less than the number of variables, depending on the relationship between the variables. The number of map layers is one of the control factors for map capacity in TMDAS.

Another factor which influences the legibility of a map is the symbol type applied to each map layer. It is obvious that the legibility of a map with the combination of a point symbol layer and a line symbol layer, would be better than a map with the combination of two area symbol layers. Therefore, the symbol type is another factor used in TMDAS. In addition, the purpose of the map, and the medium in which the map is to be found are both important factors for determining map capacity, and both have been considered in TMDAS. The mechanism for using these factors in combination to control map capacity will be introduced in detail in chapter 4.

2.4.4 Cartographic Conventions and Map Aesthetics

Convention is another factor of map design. For example, the colour green is normally used for forests, and blue for water. It would hardly be acceptable for the red colour to be used for forests on maps, although it is the colour of forests in most integrated false colour remote sensing images. In certain applications, such as geological mapping, some standardization and specifications have been established with respect to map scales, contents, symbols. The making of these special kinds of maps is not included in TMDAS.

To evaluate map aesthetics, Dent (1990) used three criteria: harmony, composition, and clarity. Harmony is viewed as the relationship between different variables. Composition refers to the graphical arrangement of variables to emphasize the main theme. Clarity deals with the ease of reading maps. Aesthetics involves all aspects of map design and is not easy to achieve in the automated environment. In TMDAS, however, particular levels of controls have been applied with respect to the last two aspects, composition and clarity. In addition to the control schemes discussed above for map capacity, the system also checks for disturbing or misleading elements when any two kinds of map types are used together, although the combination may not exceed the capacity of the map (for more details, refer to chapter 5). It is important to emphasize that none of the systems can work as well or as flexibly as human cartographers can in this respect.

2.4.5 Map Contents and Multiple Variable Mapping

Combining data to show relationships is the purpose of multiple variable mapping (Cuff and Mattson, 1982). The relationship between variables and the graphic schemes to combine the variables are the two issues in multi-variable mapping. The first is related to data properties and has been discussed in section 2.2.3, while the second is related to map properties. From a cartographic perspective, the following questions must be answered: what is the maximum number of variables allowed in each specific map? Can all of the proposed variables be represented on one map in the required medium? If not, can any data processing method be applied in order to do so? What is the best graphic design for each variable? Can these best designs be simply combined to form one map? If not, which one needs to be changed and how? In map design, multi-variable mapping is much more complex than single variable mapping. But, "it often is more productive to consider more than one data set and to devise a map showing how those variables coincide (or fail to coincide) within the study area" (Cuff and Mattson, 1982: 47). Multi-variable mapping has frequently been used in conventional mapping. The answer to the questions mentioned

above involves almost every aspect of map design. Usually, the answer is found in the hands of experienced cartographers, and is normally difficult to describe. It is thus a great challenge to simulate the human expert using machine intelligence. The methodology applied in TMDAS will be discussed in detail in chapter 5.

2.5 Cartographic Design vs. Geographic Data

In the previous two sections, we have reviewed the properties of map objects and the properties of maps. It can be summarized that map design is a procedure of *matching map properties with data properties*. To design a thematic map, one needs to know 1) the properties of the data to be mapped, i.e. the spatial distribution and measurement scales, and, 2) the relationships between the variables, if there is more than one variable involved. Based on these, then, one needs to decide 1) the map properties, i.e. the symbol types and the visual dimensions, to be applied, and, 2) the methods for representing multiple variables, if multi-variable mapping is involved. The design begins with acquiring data properties, and ends with making decisions on map properties. In this process, intensive geographic and cartographic knowledge is involved.

2.6 Summary

This chapter has discussed the knowledge dealing with thematic map design. The knowledge frame established in this chapter forms the basis of knowledge acquisition, and hence, the basis of the knowledge base of TMDAS. But as mentioned before, there are certain parts in the frame where difficulties may occur in knowledge extraction for expert system development, because not much research has been done in these areas in conventional mapping. There is certainly a need for more effort in these areas. Further studies on some of these issues, as they apply to this project, will be introduced in chapter 5.

3. ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS

3.1 Introduction

This chapter gives a brief introduction of Artificial Intelligence (AI) and Expert Systems (ES). Technical details are avoided, and, instead, a general review for non-AI specialists of both AI and ES is presented through an introduction to the basic principles and methodologies for problem solving, in order to explain the selection of this technique for the project, and its application. A historical and application review of AI and ES is given first, then the presentation focuses on ES related issues. The configuration of an expert system is discussed, and the two important issues of knowledge representation and solution search are addressed. Finally, the tools used in expert system development are introduced.

3.2 Historical Sketch and Application Review

AI, as a relatively young discipline (Luger and Stubblefield, 1989, Partridge, 1991), began with the emergence of the modern computer during the 1940s and 1950s, and has been rapidly developing since the late 1970s (Patterson, 1990). The main concern of AI is to build systems on high speed machines to emulate some human (brain) abilities. It is based on the idea that all intelligent activity can be formalized and described through some sort of computable functions, and be mimicked by a digital computer (Schutzer, 1987). The following is a definition of AI given by Patterson:

"AI is a branch of computer science concerned with the study and creation of computer systems that exhibit some form of intelligence: systems that learn new concepts and tasks, systems that can reason and draw useful conclusions about the world around us, systems that can understand a natural language or perceive and comprehend a visual sense, and systems

that perform other types of feats that require human types of intelligence."

(Patterson, 1990: 2)

Unlike conventional computer systems, an AI system is capable of not only storing and processing data, but also of acquiring, representing, and manipulating knowledge. A conceptual overview of AI is illustrated in figure 3.1. As shown in this figure, the important components in designing an AI system are knowledge acquisition, knowledge representation, knowledge manipulation (through deduction), and the search-oriented control strategy (the inference engine).

Figure 3.1 AI Conceptual Overview (from Schutzer, 1987: 14)

AI, as a whole, can also be decomposed into a number of subdisciplines by applications, although the essential approaches in problem solving applied to each of them are very similar. Some of these major applications are listed below:

Game playing: unlike a specific academic discipline, game playing is a popular activity for most people. However, games have an inherent intellectual appeal. Most of them use

well-defined sets of rules, and can generate an extremely large search space. Therefore, game playing is considered a rich field for the study of heuristic searching, and much of the earlier research in state space searches was carried out in this area (Luger and Stubblefield, 1989).

Theorem proving: automated theorem proving, one of the most fruitful branches of AI, lies in the rigor and generality of logic. "Problems can be attacked by representing the problem description and relevant background information as logical axioms and treating problem instances as theorems to be proved" (Luger and Stubblefield, 1989: 15). It forms the foundation in formalization of search algorithms and in the development of formal representation languages (discussed in detail later).

Natural language comprehension: natural language is one of the most fundamental aspects of human intelligence, a challenge to machine intelligence because of the intensive knowledge required. It is, therefore, one of the long-standing goals of AI research.

Planning and robotics: planning is a human activity involving comprehensive thinking and decision making. People often use hierarchical problem decomposition in making a plan, but it is quite difficult to write programs that break a simple task into subproblems for computers, because of the sophisticated heuristics and extensive knowledge required (Luger and Stubblefield, 1989). Planning is also the basis of robot design. In addition, an ideal robot should be able to formulate a plan with incomplete information and adjust its behavior while executing the plan. But this is still a very challenging domain in AI research.

Machine learning: learning is one of the more important aspects in AI, because the ability to learn is one of the essential components of intelligent activities. Machine learning has remained a difficult problem for AI (Luger and Stubblefield, 1989), although some

research has suggested that it is not impossible. It is believed that a breakthrough in machine learning would vastly improve the performance of expert systems.

Expert system: in the field of problem solving, research was once addressed to the development of a system which would have the ability to solve all types of problems. One of the best known systems is GPS, (General Problem Solver), developed by Newell, Simon and Shaw (Lucas and Gaag, 1991). The basic idea of GPS is to represent a general problem with initial stage, goal stage, and a set of transitions, then to generate a sequence of transitions that transform the initial stage to the goal stage. However, the system was less successful than anticipated because of its tendency to be a *general* problem solver, in which only generic knowledge was used. Generic knowledge is not sufficient in solving all real world problems. In contrast, expert system technology succeeds because it is highly associated with *domain-specific* knowledge. Some of the well-known successful systems developed at earlier stages in the research are DENTRAL, developed at Stanford in the late 1960s in the field of organic chemistry, MYCIN, developed in the mid-1970s for medicine, PROSPECTOR, for determining the probable location and type of ore deposits, and XCON, for configuring VAX computers. Many expert systems have been developed in recent years. Expert system technology has been considered one of the most successful areas of AI application.

The rest of this chapter focuses on expert systems. The principles and technologies of expert systems are introduced in more detail, because this is the technique used in the development of TMDAS.

3.3 Knowledge Representation

Knowledge is difficult to define precisely (Patterson, 1990), but it is usually viewed as a concept at a higher level of abstraction and understanding, which deals with facts and principles accumulated by humankind. Knowledge representation deals with the issues of

representation of facts, concepts, and rules in a proper form, and of their organization using proper data structures for efficient storage and use. A representational scheme should: 1) be adequate to express all the necessary information; 2) support efficient execution of the resulting code; 3) provide a natural schema for expressing the required knowledge (Luger and Stubblefield, 1989). In the following section, some of the more popular representation schemes are reviewed. Some technical terms in AI are used, but they will be introduced with examples. For precise definitions of these terms, please refer to the AI text books listed in the bibliography.

3.3.1 Logical Representation Schemes

There are a number of logical representation schemes such as propositional logic, predicate logic, and clausal form of logic, "... but the principal method for representing knowledge logically is to employ first-order-predicate logic" (Biondo, 1990). *First-order predicate* representation is the basis of PROLOG, the programming language used in the development of TMDAS. Therefore, it is discussed in detail here.

First-order predicate logic: the precise definition of first-order predicate logic can not be given without defining related terms, but it can still be understood with some examples. The following symbols are used (for detail, refer to Luger and Stubblefield, 1989, Lucas and Gaag, 1991):

Truth symbols true and false

Constant symbols, usually having the first character lowercase such as "ottawa".

Variable symbols, usually beginning with an upper-case letter such as "Map_symbol".

Function symbols, usually denoted by lower-case. Each function symbol can have a number of arguments. Examples of function symbols are

plus(2,3) – the expression has a value of 5.

Logical connectives, defined as follows:

negation:	$\neg$	(not)
conjunction:	$\wedge$	(and)
disjunction:	$\vee$	(or)
implication:	$\rightarrow$	(if then)
bi-implication:	$\leftrightarrow$	(if and only if)

Two *quantifiers*: the universal quantifier $\forall$ and the existential quantifier $\exists$. If X is a variable, $\forall x$ should read as "for each X" or "for all X", and $\exists x$ should read as "there exists an X".

A number of *auxiliary symbols* such as parentheses and commas.

Using these symbols, a so-called *well-formed formula*, WFF, (for detail, refer to Lucas and Gaag, 1991) is defined in predicate logic. Based on this, the laws of equivalence for logical connectives, and for the connectives and quantifiers are introduced as shown in Figures 3.2 and 3.3.

$\neg(\neg F) \equiv F$	(a)
$F \vee G \equiv G \vee F$	(b)
$F \wedge G \equiv G \wedge F$	(c)
$(F \wedge G) \wedge H \equiv F \wedge (G \wedge H)$	(d)
$(F \vee G) \vee H \equiv F \vee (G \vee H)$	(e)
$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$	(f)
$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$	(g)
$F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F)$	(h)
$F \rightarrow G \equiv \neg F \vee G$	(i)
$\neg(F \wedge G) \equiv \neg F \vee \neg G$	(j)
$\neg(F \vee G) \equiv \neg F \wedge \neg G$	(k)

Figure 3.2 Laws of Equivalence (from Lucas and Gaag, 1991: 26)

$\neg \exists x P(x) \equiv \forall x \neg P(x)$	(a)
$\neg \forall x P(x) \equiv \exists x \neg P(x)$	(b)
$\forall x (P(x) \wedge Q(x)) \equiv \forall x P(x) \wedge \forall x Q(x)$	(c)
$\exists x (P(x) \vee Q(x)) \equiv \exists x P(x) \vee \exists x Q(x)$	(d)
$\forall x P(x) \equiv \forall y P(y)$	(e)
$\exists x P(x) \equiv \exists y P(y)$	(f)

Figure 3.3 Laws of Equivalence for Quantifiers (from Lucas and Gaag, 1991: 33)

Almost all grammatically correct English sentences can be represented by the method described above. Some simple examples are found below:

"All maps have projections." $\rightarrow \forall X \text{ map}(X) \rightarrow \text{has_projection}(X)$

"Some features are shown in point symbols."

$\rightarrow \exists X (\text{feature}(X) \wedge \text{shown}(X, \text{point_symbol}))$

The advantage of representing knowledge in a formal representation, like the formulas on the right side above, is that known facts can be used to derive other facts that must also be true. Furthermore, the truth of any statement can be checked. For example, we already have the fact that "all maps have projections" above. Now, if we add another fact that "all topographic maps are maps", that is

$\forall X \text{ topo_map}(X) \rightarrow \text{map}(X)$

which reads "for any object X, if X is a topographic map, then X is a map", we can then infer that the following fact must be true (to do this kind of reasoning, logic has to follow a set of rules, so-called *inference rules*, which are not discussed here):

$\forall X \text{ topo_map}(X) \rightarrow \text{has_projection}(X)$ ("all topographic maps have projections")

Theoretically, the knowledge base established in this manner can be kept logically consistent and all conclusions can be guaranteed to be correct. But, in practice, this can be very difficult, because "The key to using first-order predicate logic successfully in a knowledge base is to accurately represent the knowledge base on the basis of formal symbolic representations and interpretations and that may not be easy" (Biondo, 1990: 48).

3.3.2 Production Rules

Production rules constitute the characteristic elements of a modular knowledge representation scheme known as a production system. This is the most popular knowledge representation method used in expert systems, because it facilitates human understanding and easy modifications of systems. The knowledge base consists of rules, called production rules, in the form of condition-action pairs, as:

IF this condition occurs, THEN do this action

In production systems, encoding human expertise in the form of rules, and designing pattern-driven search algorithms are the tasks central to the design of rule-based systems.

3.3.3 Semantic Networks

Semantic networks are another popular scheme for representing knowledge in a more structured fashion. Many various semantic network systems are currently in use. The basic concept is to use directed graphs to describe the properties of objects, events, concepts, situations or actions, and their relationships. Figure 3.4 is a simple example for representing the properties of data and its relations. Property inheritance is one of the main characteristics of this representation. As in the given example, "primary industry" is one of the subsets of "total output", therefore, the former will inherit the properties, point spatial dimension and absolute quantitative measurement scale, of the latter. This representation

method is used to depict the relationships between variables in establishing the dynamic data base in TMDAS, and will be introduced in chapter 6.

Figure 3.4 An Example of Semantic Networks

3.3.4 Frames

A frame is a structure for organizing knowledge. It consists of a series of *slots*, each of which represents a standard property or attribute of the element represented by the frame. For complex problems, slots can be further divided into subslots, or be filled by lower-level frames. A frame system uses interrelated frames to represent a problem domain. The same problem domain represented by the semantic networks shown in figure 3.4 can be structured by frames as shown in figure 3.5. In general, frames are complex data structures and they are preferable for stereotyped objects.

3.4 Solution Search

Solution search is considered the universal problem-solving mechanism in AI. Any problem solving can be regarded as a solution search procedure. The sequence of steps required for the solution of problems is not generally known *a priori*, but must be determined by trial-and-error examination of the alternatives (Biondo, 1990).

Figure 3.5 An Example of Frames

3.4.1 State Space Search

State space search is a technique often applied in computer programs. Based on graph theory, it can be used to analyze the structure and complexity of both a problem and the procedures used to solve the problem (Luger and Stubblefield, 1989). As shown in figure 3.6, the graph consists of a set of nodes and a set of arcs, nodes corresponding to partial problem solution *states* and arcs to steps in a problem-solving process. Then, a state space can be represented by four tuples $[N, A, S, GD]$, where (Luger and Stubblefield, 1989):

N , the set of nodes or states of the graph;

A , the set of arcs between nodes;

S , a non-empty subset of N , contains the start state(s) of the problem;

GD , a non-empty subset of N , contains the goal state(s) of the problem.

A *solution path* is a path through the graph from a node in S to a node in GD . State space search characterizes problem solving as the process of finding a solution path from the start state to a goal (Luger and Stubblefield, 1989).

Figure 3.6 An Example of State Space Representation

3.4.2 Direction of Search – Forward Chaining and Backward Chaining

A state space can be searched in two directions: from the known data toward the goal, or from the goal back to the data. The former is referred as *forward chaining*, or *data-driven search*, and the latter as *backward chaining*, or *goal-driven search*.

In forward chaining search, the search starts with a set of given facts and rules. By applying the rules to the facts, new facts are generated. This procedure is repeated until the goal is reached. Meanwhile, the solution path is generated.

In backward chaining search, the search starts with the goal, determines the rules which can generate that goal and checks what conditions must be true to use the rules. These conditions will be considered subgoals. This procedure is repeated until all the subgoals are found for the given facts. Following is an example of these two systems:

```

fact1: point_feature(ottawa)
rule1: IF point_feature(X) THEN point_map(X)
rule2: IF point_map(X) THEN point_symbol(X)
Goal: point_symbol(ottawa) ?
  
```

Forward chaining:

step1: point_feature(ottawa) (fact1)
step2: point_map(ottawa) (rule1)
step3: point_symbol(ottawa) (rule2)
proven!!

Backward chaining:

point_symbol(ottawa) (rule2)
point_map(ottawa) (rule1)
point_feature(ottawa) (fact1)
proven!!

As can be seen, both data-driven and goal-driven problem solvers search the same state space graph, but the order and actual number of states searched can be very different if the problem is complex. The strategy used depends on the problem properties, which include the complexity of the rules, the "shape" of the state space, and the nature and availability of the problem data (Luger and Stubblefield, 1989).

Backtracking: in both searching procedures, there is no infallible control for the choosing of a solution path, and dead ends can frequently be encountered. Actually, it is the same with human beings. To solve a problem, a person may have several solutions. If one fails, another will be tried. Backtracking is employed by computers for the same reason. The search is carried out and pursues a path until it reaches either a solution or a dead end. If it reaches the solution, it retains the solution path. If it reaches a dead end, it backtracks to the most recent node on the path which has unchecked siblings. The search continues from there.

3.4.3 Ordering of States – Depth-First and Breadth-First

In addition to forward chaining and backward chaining, a search algorithm must also decide the order in which states are examined. There are two commonly used strategies named *depth-first* search, which bypasses a state's siblings and goes as deep into the search space as possible, and *breadth-first* search, which searches the space level-by-level. The difference can be seen from figure 3.7.

Figure 3.7 Depth-First vs. Breadth-First Search

3.4.4 Heuristic Search

In state space search, *heuristics* are defined as "formalized rules for choosing those branches in a state space that are most likely to lead to an acceptable solution" (Luger and Stubblefield, 1989: 149). Heuristics are used in two situations: a problem may not have an exact solution due to the inherent ambiguities in the problem statement or insufficient available data; or, a problem may have an exact solution, but the cost of computation may be excessive. Heuristics are fallible because they are often based on experience, and informal guesses of the next step to be taken in problem solving. Heuristics are usually applied to expert systems in the form of certainty factors, which attach a numeric weight to the conclusion of each rule together with algorithms for calculating the certainty factors.

3.5 Expert System Architecture

An expert system can be defined as a computer program which is based on knowledge and reasoning techniques, and which can offer "expert quality" solutions to problems in a specific domain (Luger and Stubblefield, 1989, Schutzer, 1987). Like human experts,

expert systems tend to be specialized, focusing on a narrow set of problems, and their knowledge is both theoretical and practical. Unlike human beings, however, their knowledge is not gained from experience, but extracted from humans and encoded in a formal language (in advanced systems, machine learning is included as well). Actually, knowledge acquisition is, in general, the major task in building expert systems and it is the main focus of the development of TMDAS, in particular.

Expert system development is also termed *knowledge engineering*. In building an expert system, three kinds of people are normally involved: the *knowledge engineer*, who is the AI specialist, and is familiar with knowledge representation and AI programming languages; the *domain expert*, who can provide knowledge at the expert level in the problem domain; and the *end user*, who uses the system, often defines design constraints, and performs the system evaluation. Cooperation between them is the basis of success in expert system development.

Because of the distinct characteristics of expert system development, expert system programming is often referred as *exploratory programming*, and is required to be: open to inspection (which means that the user can, at any time, inspect the state of the system reasoning), easily modifiable, and should employ heuristics (using heuristic problem-solving methods) (Luger and Stubblefield, 1989). Figure 3.8 is a general architecture of expert systems. Each of the modules is briefly introduced below.

Knowledge base is the core of the system in the architecture. It contains domain-specific knowledge represented in a specific format. The knowledge base is divided into two parts, the general knowledge base and the case-specific data. The former is the permanent knowledge stored in the system, and the later is often used as a dynamic data base established at run time to store traced facts and draw conclusions in the case under consideration.

Inference engine is the interpreter for the knowledge base. It consists of algorithms for manipulating the knowledge, and performs the recognize-act cycle (for detail, refer to chapter 6). Unlike in conventional systems, this part is separated from the knowledge base to give flexibility to the updating of the knowledge base.

Figure 3.8 Expert System Architecture (from Luger and Stubblefield, 1989: 294)

Explanation subsystem is a subsystem to explain the reasoning to the user. It normally includes: **HOW queries**, that is, the justifications for the conclusion given by the system; **WHY queries**, that is, the explanation of why a specific piece of data is needed; and, in some of the cases, tutorial explanations to help in the use of the system.

User interface is a subsystem to build a friendly working environment for the user by employing different kinds of techniques such as questions, menus, or graphics.

Knowledge base editor is optional for the end-user system, and is used to build and update the knowledge base. Access to the knowledge base is normally restricted to system development purposes only.

The implementation of each of these modules in TMDAS is discussed in chapter 6.

3.6 Languages for AI Problem Solving and Expert System Shells

Based on the analysis of the inherent characteristics of AI problem solving, the following are the requirements for AI programming languages:

- Support of symbolic computation;

- Flexibility of control;

- Support of exploratory programming methodologies;

- Late binding and constraint propagation;

- Clear and well-defined semantics (for details, see Luger and Stubblefield, 1989).

Two of the most commonly used languages in AI programming are PROLOG and LISP. Both of them have distinct characteristics which distinguish them from traditional programming languages such as Pascal, C, or FORTRAN.

PROLOG

PROLOG stands for PROgramming in LOGic. It is based on first-order predicate calculus. Unlike conventional programming languages, which are referred as *procedural* languages, PROLOG is a *declarative* language. PROLOG programs, as the examples shown below,

```
line_feature(queensway).
```

```
line_symbol(X):- line_feature(X).
```

are collections of descriptive facts about a domain and rules for deriving conclusions from those facts. Prolog uses unification, a search algorithm for determining the substitutions

needed to make two predicate calculus expressions match (Luger and Stubblefield, 1989). Prolog has frequently been used in expert system development, and it is also one of the languages used in the implementation of TMDAS. Some details on Prolog are discussed in chapter 6.

LISP

LISP is an acronym for LISt Processing. It is also a very popular language in AI. In contrast to Prolog, LISP is procedural; it is different from traditional languages, since LISP is functional: its syntax and semantics are derived from the mathematical theory of recursive functions. The power of functional programming combined with a powerful set of high-level tools for building symbolic data structures are important aspects of LISP. In LISP, the list is the basis of both programs and data structures. The symbolic elements of LISP are symbolic expressions, or s-expressions. An s-expression can be either an atom or a list (for details, see Luger and Stubblefield, 1989). For example:

100

x

(- (+3 4) 7)

(on (book table))

Like PROLOG, LISP is a popular language for AI applications. It is difficult to determine which language is better. The selection may depend on the experience of the knowledge engineer. Using the programming language as the tool for development can give great flexibilities in system design, especially when the system must be integrated with a conventional data base, or with other systems. This is one of the reasons why programming languages are used in the development of the prototype systems instead of expert system shells.

Expert system shell:

Although languages are powerful and flexible, in order to develop an expert system by using these languages directly, one has to pay a disproportionate amount of attention to the implementational aspects of the system, which have nothing to do with the domain of the knowledge. Besides, the expert knowledge and the algorithms for applying this knowledge will be highly interwoven and not easy to separate. This will make knowledge update difficult. To make the development easier, special development tools, called *expert system shells*, are available. Unlike a complete expert system, a shell does not have a knowledge base, but, instead, it provides a knowledge base builder's interface that helps the user build a knowledge base. The idea behind this is that the user can produce a true expert system for any problem domain by "filling" the shell with the specific domain knowledge required.

The problem with using shells is the loss of flexibility, because shells are often the by-product of earlier successful applications, and they are more suitable to the domains related to those applications. Considering the development needs of TMDAS, a programming language was selected for better flexibility, yet it was also considered that it is necessary to separate the domain knowledge from the algorithms for reasoning, because of the reasons mentioned above. Therefore, the approach applied in the implementation of TMDAS was to develop an expert system shell first using the Prolog language. The shell will meet all requirements of the project, while the knowledge base will include domain related knowledge only. The implementation of TMDAS will be discussed in detail in chapter 6.

3.7 Summary

In this chapter, expert systems and some aspects of AI related to the development of TMDAS have been introduced. The issues of knowledge representation and solution searching have been addressed. Expert system development tools, both AI programming

languages and expert system shells, have been discussed. The technology introduced in this chapter lays the foundation for the implementation of TMDAS.

4. THE CONCEPT DESIGN OF TMDAS

4.1 Introduction

Thematic map design is a complex problem domain. The development of a fully operational thematic map design advisory system is a long term engineering project. An incremental approach needs to be applied toward the development of such a system, and the design should be highly modular. The concept design, therefore, becomes important to ensure final success, and has to be done carefully at the initial stage. The main issue that the concept design should deal with is how to divide the whole procedure into modules in such a way that the development of each module provides a step towards the final goal, and, secondly, that all the modules can be combined into the evolving production system. A good concept design is based on a detailed analysis of user requirements, and a thorough understanding of the characteristics of the problem domain. These are the issues discussed in this chapter. In addition, an optimal framework, as well as a specific framework that has been modified for TMDAS, are presented. Each logical module developed in TMDAS, its function, and its relationships with other modules are introduced.

4.2 User Requirement Analysis and Problem Identification

User requirement analysis is an important procedure in system development. A clearly defined user requirement is useful in the whole procedure from problem identification through task decomposition, in every stage of system development. This is especially true for expert system development, because it has been proven that a clearly defined user type as well as problem domain boundary are crucial towards the success of the system.

It has been argued, in the development of expert systems for thematic map design, that "the expertise needed by the users of a thematic mapping package covers two main aspects, i.e., professional knowledge in the field to be mapped and cartography" (Wang, 1990: 782). It

is these two types of knowledge that are needed in thematic map design. The issue, however, is: should both types of expertise be provided by the system. This is a key question, because it will affect the entire concept design. Since professional knowledge in the field is domain dependent, if this type of knowledge is included in the system, the system must also be domain dependent. On the other hand, if this domain dependent knowledge is separated from the body of knowledge for thematic map design, the remaining aspect will be cartographic knowledge, which is domain independent. If only this cartographic knowledge is included, the system should be able to deal with a variety of thematic mapping purposes in a generic manner. To answer this question, we need to clarify the purpose of the system, and to identify the potential users.

The purpose of the development of TMDAS, as stated in chapter 1, is to help cartographically untrained users to map their data, using correct graphic representation methods, in an automated mapping environment. The potential users of TMDAS are from different mapping application domains, such as geography, surveying, environmental research, planning, etc., they need to map their data, and usually accomplish the mapping using GIS or thematic mapping software packages. In other words, they have the first type of knowledge mentioned above, but not the second. The assumption on the type of user, therefore, can be described as people who have expertise in their domain, hence have knowledge of their data, and want to produce maps for specific purposes, but do not have knowledge on cartographic representations. This assumption is important in the entire concept design. Its effect can be seen from the discussion of the two different approaches introduced in the following sections.

Based on the preceding analysis, it can be concluded that the potential user can come from different mapping application domains, and be without knowledge of cartography. Therefore, there will be two criteria that the system should satisfy. One is that the goal of the system should be toward the support of all types of thematic mapping requirements,

that is, the system should work domain-independently. The second criterion is that the cartographic advice given by the system should be understandable by a user who has no cartographic knowledge. There should be, then, a communication mechanism for users to interact with the system for written or graphic explanations. A consulting procedure can be a learning procedure for users.

4.3 Approaches in Problem Solving

In the natural world, there is often more than one method in solving a problem, and the same is true in expert system development. It is, therefore, a challenge to find the proper approach in problem solving. The following discussion will introduce two different approaches which may be applied to the development of a thematic map design advisory system which fulfils the functions described above. The different mechanisms, advantages and disadvantages of each approach will be analyzed.

4.3.1 Theme Oriented Approach

The theme oriented approach is the approach applied in most of the prototype systems in thematic map design (see Laurema, et al., 1991, and Wang, 1990). This type of approach is designed to include both professional knowledge of the mapping field and cartographic knowledge. As Wang described it, the system should meet two goals:

- "1. User enters a mapping theme, the system can find a set of proper information as response.
2. User selects information, the system can tell what meaning the information shows, and if necessary more related information will be provided" (Wang, 1990: 783).

In this approach, the system functions like the integration of a cartographer and an expert in a specific mapping domain. Based on a specific mapping requirement, the system can tell the user directly what kind of data should be used and what type of map should be made.

The advantage of this kind of system is that little knowledge is required from the user. However, the main disadvantage is that the system is domain dependent. The cartographic advice is available only if the mapping requirement is within a specific domain and within a predefined map list. This limits the use of the system to a small group of users, because even within a domain, there are always new themes to be mapped which are not included in the knowledge base.

One argument to solve this problem is to build the system module by module in order to increase its capacity gradually. Laurema (1991) stated that "we have used a 'bottom-up' approach in this study. In other words, we started by analyzing a restricted special case, the presentation of land-use classification data. Once the problems related to this special case have been solved, we can seek to extend and modify the methods for general application" (Laurema, et al., 1991: 11). The issue here is the extension of the system. Theme related knowledge is the knowledge used to describe a particular theme. For example, the knowledge base for the theme of population can include knowledge such as the size of the population, the population density, and the proper graphic representation methods for each of them. This kind of knowledge can be called specific knowledge, and always depends on particular themes. This means that if a theme is to be covered by the knowledge base, specific knowledge for that theme must be extracted, which may lead to a theme oriented module. The following two problems, therefore, can arise: 1) should the final system be an integrated system? 2) What would the interface between different modules be? There are two main obstacles to this theme oriented approach: 1) there is no clear boundary between the different themes; and, 2) no matter the size of the knowledge base, there will always be some mapping requirements outside the knowledge base, because the number of objects which can be mapped is infinite. Therefore, it can be concluded that this approach is good if the system is for specific purposes, but may not be optimum for a generic thematic map design expert system. Since the latter is the purpose of this project, it is necessary to seek another approach.

4.3.2 Cartographic Function and Data Property Oriented Approach

The basic premise of the approach proposed in this thesis consists of three aspects: 1) divide the whole mapping procedure into modules by *cartographic functions*, or, in other words, divide the cartographic activity into steps; 2) define the input and output of each module and develop the system module by module; and, 3) carry out the reasoning for each module based on generic *data properties*.

Figure 4.1 and 4.2 are the illustration and comparisons of the two approaches. The main advantages of cartographic function and data property oriented approach, in contrast to the theme oriented approach, are: first, the mapping procedure can be divided into a limited number of modules, while in the theme oriented approach, the number of themes is unlimited; second, with clearly defined input and output, the final system can be built module by module, while in the theme oriented approach, the interfaces between the modules are difficult issues. To apply this approach, however, two questions must be considered. The first is whether the generic data property is sufficient to drive the inference of every module, while the second is the method to divide the modules. A thorough understanding of the ways in which a human cartographer handles a design problem is required for both of the questions. The first question will be discussed in this section, because it actually determines the feasibility of this approach, and the second question, the application of the approach will be discussed in the next section.

Some of the analysis of the feasibility of using data properties in map design was carried out in chapter 2 in the introduction to data properties and in the user requirement analysis sections in this chapter. As stated in these sections, in conventional map design, a human cartographer does need to have some knowledge of the data to be mapped in order to make

Figure 4.1 Theme-Oriented Approach

Figure 4.2 Cartographic Function and Data Property Oriented Approach

a proper map design, but the question is the amount of knowledge the cartographer must know about the specific domain. A human cartographer may be asked to make many different maps. It is not possible for one cartographer to be an expert in all the domains involved. In fact, as Robinson pointed out, "there is likely to be more difference between a large-scale map of surface bedrock in an area and a small-scale map of plate tectonics in the Atlantic basin – both geological maps – than there would be between a soil map and a vegetation map of a given area. Cartography is independent of subject matter" (Robinson, et al., 1984: 11). Based on the analysis given before, cartography is *primarily dependent* on data properties: spatial distribution (which is related to map scale, see chapter 2), and measurement scale. In addition, for a high quality thematic map design, cooperation between experts in a mapping domain is usually necessary. Knowing the data properties and mapping purposes, the cartographer should be able to formulate the design, which is, as described in chapter 2, to match the map properties (which consist of symbol types and visual dimensions) with data properties. It can be concluded, then, that it is possible to carry out thematic map design based on generic data properties, assuming the user of the system has knowledge of the mapping domain.

The cartographic function and data property oriented approach simulates the human cartographer in solving map design problems. The purpose of this approach is to develop a system to take over some of the functions of the cartographers producing thematic maps. The main disadvantage of the approach is that it requires that the user has some knowledge of the data to be mapped. The system is *not* designed to be used by *anybody*. But, based on the user requirement analysis, this should not be a serious problem, since most GIS users are experts in their domains. Moreover, there is also a positive aspect to this restriction, that is, it emphasizes that map making is a professional activity. If one wants to produce a map, one must know the purpose of the map, and must have some knowledge of the data. Otherwise, the quality of the map can not be guaranteed, even with an expert system. Making maps without understanding the data should not be encouraged. The

cartographic function and data property oriented approach has been used for the development of TMDAS.

With this restriction on the user's side, the system may work somewhat like the amplified intelligence approach presented by Weibel and Battenfield (1988), which incorporates elements of an expert system with the flexibility of human interaction, so that the computer system solves a given problem in teamwork with a human operator (Weibel 1991).

4.4 System Functionality and Modeling

Based on the approach mentioned above, the other issue to consider, is the decomposition of the map design procedure into modules. There are many alternatives for the deconstruction, and they can affect cartographic knowledge modeling. The following is the series of steps followed by a human cartographer creating graphic design with a five-variable data set. The first step would be the identification of the data properties (e.g. point or line features, nominal or ordinal data, and their relationships), and of the purpose of the map. Then, the cartographer would begin planning the proper graphic method to be used for each variable (or group of variables for compound symbols), keeping in mind the graphic effect on all other variables. For example, method A may be the best for variable 1, but since method B has to be used for variable 5, then method C should be used for variable 1 because of the final result, etc. After several cycles, the cartographer will arrive at a compromised comprehensive representation scheme comprising symbol types (e.g. point or line symbols) and visual variables (e.g. form or size) for all variables (experienced cartographers may reform the cycles mentioned above simultaneously). Finally, the cartographer will test the scheme with graphics, and decide on specific symbol characteristics such as the hue to be used, the range of the value or size, etc. This kind of test may need to be repeated several times until a satisfactory result is achieved. The last step is a procedure in which visual perception and comprehensive analysis of the graphics are required, though the exact steps and sequences may change from one expert to the next.

Based on this analysis, the concept design of an optimal framework and a modified framework are given in figure 4.3.

4.4.1 An Optimal Framework

Figure 4.3.a is the framework of an optimal thematic map design system, assuming that GIS/EMS (Electronic Mapping System) and ES (Expert System) are highly integrated. In this framework, the digital data and mapping objectives would be stored in a GIS as basic elements on which the map design would be based. The *information interpretation* module would be able to access the GIS data base directly to extract information about data properties and mapping objectives. Then, *map type selection* and *map type combination* modules would be used to determine the suitable map type for each variable, and to check the possible combinations of selected map types for all variables, according to their relationships. Each map type is associated with a symbol type which must be used. The *visual dimension selection* module would then recommend the appropriate visual dimensions to be applied under each map type. The *statistical data analysis* module would be used when data processing was needed (e.g. data classification). With the output from previous modules, the *actual visual dimension design* module would be applied to complete the graphic design. In the last module, a highly integrated ES and GIS environment is required, because, in this module, not only textual rules, but also "graphic rules" are needed – testing and modifying of the graphics, based on the graphics produced automatically, is an unavoidable task, even for the most experienced cartographers, in order to achieve an artistic expression of the information portrayed. This module, therefore, is under both ES and GIS environments, which implies that access to the inference engine, the spatial data base, as well as the graphics functions are needed simultaneously. This framework represents an optimal configuration.

Figure 4.3.a Optimal Framework for Automated Map Design

Figure 4.3.b Modified Framework for Automated Map Design

4.4.2 The Modified Framework

Figure 4.3.b is a modified framework and is used in this project due to the fact that the total integration of ES and GIS is beyond the scope of this study. Three changes have been made to the optimal framework: i) the GIS-ES interface is changed to the System-User

interface; ii) the information interpretation module is changed to the information query module; and, iii) the actual visual dimension design module is placed in the GIS environment.

4.4.2.1 Information Query Module

This module is used to acquire data properties and map objectives on which the map design is based. Data properties consist of spatial dimensions, measurement scales, and relationships between the different variables, if more than one variable is specified. Map objectives influence map design. For example, the screen-based map should be different from a paper-based map in the complexity of the maps and in the selection of visual variables. Three levels of user interface are designed in this module. The first level is for use with data available in NAIS. The knowledge of data properties of NAIS data is extracted and stored in the knowledge base. Therefore, there will be fewer questions asked to NAIS data users. The second level is for use with any other user-brought data sets. In this case, the user's knowledge of the data is required. The third level is for use of a combination of the first two. An abstracted generic entity-relationship between variables is used to define relationships (e.g. A consists of B, or A is part of B, for details, refer to chapter 5). Based on these relationships, the variables should be grouped into map layers (as described in chapter 2), in order to form the basic graphic units for the map design procedure.

4.4.2.2 Map Type Selection Module

The map type selection module is used to identify possible map types for each map layer based on the data properties obtained from the information query module. Since the final map may be composed of several layers, the task of this module is to find the best graphic representation, termed map type, for each single layer which may consist of one or several elements, called variables. Map type selection is an essential intermediate procedure in

TMDAS, although map type is excluded from the final recommendation which consists of two components: symbol types and visual dimensions. The classification of map types is a very important aspect, which must fulfill the following functions: 1) assist in finding the proper symbol type for given variables; 2) assist in checking the possibility of combining multiple layers on one map; and, 3) assist in identifying alternative map types, when needed, based on the relationships between map types. The taxonomy of map types used in this project will be discussed in chapter 5. This module will be used after the information query module, or at any stage when a change of map type for a specific layer is required.

4.4.2.3 Map Type Combination Module

The variables have been grouped into map layers according to their relationships, and the appropriate map types for each map layer have been determined in the first two modules. Then, the result of combining these single layers into one map sheet needs to be checked, based on the analysis of the factors which influence the map capacity. For instance, it is not possible to combine three choropleth map layers into one legible map, but it is possible to combine three layers if they are dealing with point symbols, line symbols and area symbols respectively. The purpose of this module is to check the possibilities of combining all layers with their selected map types into one map sheet. If the combination is allowed cartographically, the module will further check if there are any restrictions or conditions which need to be applied. For example, if hue was used by one area symbol layer as the visual variable, it would be better to use form or orientation rather than hue on another area symbol layer in which the symbols may overlap with the first one, because the combination of two hues creates a third hue which may not be interpreted properly. If the combination is not considered suitable, the module will check the second best map type of each variable in turn, until an optimal combination scheme is reached. If the user's

requirements cannot be satisfied, the explanation and/or suggestions for possible processing which can be applied to specific variables will be given.

Capacity of the map, as discussed in chapter 2, is a complex factor influenced by many elements. It is a difficult issue in automatic mapping. In most prototype systems, there is often no mechanism to check and control the capacity of the map, instead, a fixed number of variables is normally used. For example, in ESSYD developed by Wang, it is defined that "two thematic components are considered to be the upper limit, and the only exception is made for multiple absolute information which can be shown by compound symbols, such as bars or pie graphs" (Wang, 1992: 79). Therefore, in ESSYD, the different types of map configuration are defined, including the number of variables such as "single nominal, "single ordinal", "double derived quantitative", or "1 nominal + 1 ordinal", etc. (see Wang, 1992: 79-80). And in that system, "the data can refer to either points or to areas, so there are 28 types of configuration in total" (Wang, 1992: 80). The advantage of this method is that the reasoning procedure is much simpler – it needs only to check the mapping requirement against these 28 types (in this case). But the disadvantage is that the total available map types are very limited. The crucial problem here is choosing the maximum number of variables permitted in the system (in ESSYD, it is two). Because the number is fixed, a conservative number has to be used in order to guarantee that the system will work in all cases. But in some of the cases, it will appear to be restrictive. For example, in reality, many maps often contain more than two variables. The capacity of the map is not related just to the number of variables, but also the relationship between the variables, and the graphic form used for them. The method presented here is more complex, not only because it involves two stage reasoning, but also because a repetitive check is normally needed, as shown in figure 4.3.b. and some extra concepts (such as map layers, and the relationships between variables) also have to be defined. However, by separating an integrated map into single layers and establishing two modules for single layer type selection and combination respectively, the use of a fixed number can be

avoided. Therefore, there is no predefined limit, and a wide selection of graphics will be available. In addition, the system will also check if the final recommendation is best not only for each variable, but also for the graphic combination as a whole in order to eliminate disturbing or misleading combinations. Whenever any single map type changes for some reason, this procedure needs to be repeated.

4.4.2.4 Visual Dimension Selection Module

As discussed in chapter 2, visual dimension selection is an important aspect of map design. There are certain graphic rules associated with visual perceptions that must be followed. Non-cartographers can easily make misleading maps by applying the wrong visual variables. This module will be used to identify the visual variables and constants (see chapter 2) for each variable, based on the measurement scales of the data, the map types selected from the previous modules and the perception properties of the visual variables. In addition to the basic rules, there are also some other factors which affect the selection of visual dimensions, such as the medium on which the map is based, e.g. paper or screen display. Furthermore, there are certain restrictions caused by layer combinations. Based on data properties and their relationships, the two kinds of visual variable combinations, redundant combination and "meaningful" combination, increase the capacity and legibility of the map. As a result, visual variables and constants for each layer must be worked out from this module, when more than one choice is available, the system will be able to list all of them in a best-first order.

4.4.2.5 Actual Visual Dimension Design Module

From the above four modules, the system will be able to determine the symbol types and visual dimensions to be used for the given data set. The remaining stage is to accomplish the graphic design according to this expert recommendation.

Figure 4.4 Two Point Maps

In the modified framework, this module is placed in the GIS environment (see figure 4.3.b), because, in this stage, visual and artistic aspects play an important role, thus several interaction cycles are needed before the final decision on cartographic design can be made. For example, if the variable "size" is selected for a point symbol map, the range of the symbol sizes (minimum, maximum) must be decided before the actual symbol design can be carried out. It is very difficult to extract generic written rules for the range of sizes which can be applied to all cases, because the proper size of symbols does not depend only on the number of points and the size of the mapped area, but also on the spatial configuration of symbols. This can be seen from figure 4.4, where, on the left, the symbols do not overlap, but on the right, they overlap. In cartography, the example on the right is the most common case. It should be the symbol's distribution and its relation to other map features – in other words, visual perception – that determines the size of symbols. The actual visual dimension design procedure involves complex issues of art and visual perception, and adjustment based on the graphics is needed. The actual symbol design will have to be carried out in the interactive graphic environment with the support of graphic functions and the spatial data base. Because of this, the module is not included in the prototype system.

4.5 User Interface and Explanation

Facility of explanation is one of the distinctive characteristics of an expert system. When Fisher and Mackaness (1987) asked "Are cartographic expert systems possible?", they noted that none of the published accounts of cartographic "expert systems" were able to explain the reasons that decisions were reached, and thus failed to qualify as true expert systems (Buttenfield and Mark, 1991: 134). Because expert systems are designed to produce satisfactory results, generally based on incomplete information, they could produce incorrect results that should not be blindly accepted by the user. The explanation facility, therefore, should be available to the user, in a convenient form. For TMDAS, three types of explanations are designed: WHY, EXPLAIN and HOW. WHY and HOW are expert system terminologies, and there are restricted formats to answer these questions in expert systems. When the system asks a question to the user, the user has the right to know "why do you ask me this question?"; that is what WHY means. When the system reaches a conclusion, the user should be able to ask "how did you reach that conclusion?"; that is what HOW means. The answers to both questions should not be something pre-stored in the system, instead, they must be based on dynamic inference (to qualify as true expert systems). For example, the answer to HOW must be the trace of the actual *solution path*. EXPLAIN is relatively simple. It is used to give further information related to the question, in case the user does not understand the question.

In an expert system environment, there are often two kinds of interfaces involved, one for the end user, and the other for the knowledge engineer. The first type is discussed here. To offer the end user a convenient interactive working environment, menu and window operations should be implemented. Furthermore, based on the expert recommendation given by the system, a graphic user-interface should be developed to explain the recommendation graphically, because it is not valid to assume that all the terms (such as hue, value) used in the recommendation have the same meaning to all users. Actually,

"...words are a foreign language to graphic communication. Furthermore, the words that we must use do not have specific meanings out of context" (Robinson, et al., 1984: 142). A graphic interface, therefore, should be included as a very important part of the cartographic expert system.

4.6 Summary

In this chapter, the issues related to concept design have been discussed. Based on the analysis of user requirements, and the analysis of advantages and disadvantages of the theme oriented approach, the use of the cartographic function and data property oriented approach is proposed. The feasibility of the approach and the application of the approach have been discussed through the analysis of the method used by a human cartographer to solve a design problem. Two frameworks for the selected approach were then presented. One is an optimal framework, requiring highly integrated GIS and ES. This framework can be considered as the long term goal. The second framework is a modified one which allows GIS and ES to be loosely coupled. The modified framework is designed in such a way that, it is easier to accomplish and easier to upgrade it to fit into the optimal framework. The modified framework has been implemented in the prototype of TMDAS. The basic principles of each module and the interrelationships between them have been discussed. The concept design described in this chapter governs the basic principles used in knowledge acquisition and system implementation.

5. KNOWLEDGE ACQUISITION

5.1 Introduction

Knowledge is usually viewed as a concept at a high level of abstraction, dealing with facts and principles of how facts related to each other are accumulated by humankind. Knowledge acquisition is a procedure for "the transfer and transformation of knowledge from the forms in which it is available in the world into forms that can be used by a knowledge system" (Gruber, 1989: 3). In other words, it is a procedure of transformation of knowledge from a nonoperational form, such as verbal or written natural language, into the computational framework of a computer program, which is referred to as operationalization. Knowledge acquisition is always considered as one of the greatest bottlenecks in building knowledge-rich systems (Patterson, 1990), because in most natural domains, "it is not valid to assume an already-operational and complete domain theory that includes the expert's statements as a special case. Thus operationalization will produce computational formulations of expertise that may be unfamiliar to the domain expert" (Gruber, 1989: 130). The following comparison illustrates the fundamental difference between human knowledge and formal knowledge .

The characteristics of	
human knowledge	formal knowledge
- Gained through experience	- Programmed, not learned
- Not necessarily symbolic	- Symbolic
- Often highly compiled and difficult to articulate	- Must be represented exactly
- May be inherently fuzzy and inexact	- Organized into individual rules

Figure 5.1 Fundamental Differences between Human and Formal Knowledge

To transfer human knowledge into formal knowledge, therefore, is always one of the difficult issues in knowledge engineering. It is especially true in the development of cartographic expert systems. As Laurema pointed out from his experience, although there are some basic principles which can be referred to in map design, they are mainly presented in textual form, in tables and in pictures which are not precise enough to be applied as such to knowledge-based systems (Laurema, et al., 1991). Hence, knowledge acquisition is central to this project.

The basic knowledge review and codification for establishing a knowledge frame for the development of TMDAS was presented in chapter 2. But, as noted before, there is less knowledge available in some parts of the frame, and further research needs to be done to build a complete body of knowledge for TMDAS. Moreover, all the knowledge must be transformed into formal knowledge for system implementation. Therefore, some aspects must be considered: 1) the knowledge needs to be organized and structured in order to be transferred into formal knowledge representation; 2) the knowledge needs to be represented in an easy to implement form; 3) further research is needed in the areas where mature knowledge is lacking in order to acquire the knowledge needed for the system. These issues will be discussed in this chapter.

5.2 Knowledge Structure and State Space Representation

Map design is highly abstracted intelligent work. To represent the knowledge related to this work in a formal knowledge base, problem decomposition and knowledge structuring are necessary. Figure 5.2 is the logical module of the knowledge structuring based on the TMDAS concept design. This figure also shows that in TMDAS, data properties drive the entire map design.

State space representation is a method to represent a problem based on graph theory. It has been introduced in chapter 3. To assist problem decomposition and knowledge acquisition,

Figure 5.2 TMDAS Knowledge Structure

the problem solving in map design can be represented using the state space representation method. Figure 5.3 is a simple example. In this graph, nodes correspond to partial problem solution states, and arcs correspond to the steps in a problem-solving process. The solution path is composed of the arcs connected to represent the problem solving path high-lighted in the figure. Through the use of this type of graph, problems can be decomposed. Therefore, state space representation is useful in defining the knowledge in detail and in system implementation.

Figure 5.3 An Example of State Space Representation of Symbol Design

5.3 The Design of Knowledge Acquisition Tables

Knowledge acquisition can be carried out by interviewing domain experts, studying existing materials, applying automatic knowledge acquisition technologies, or through some combinations of all these methods. Knowledge acquisition is rather time consuming.

and the decision on the methodology to be used must often be made according to the specific situation.

Knowledge acquisition in TMDAS is mainly based on some authoritative textbooks, as well as on the analysis of the design of selected maps from the Fifth and Fourth Edition of National Atlas of Canada. Based on the concept design and knowledge structure, a series of knowledge acquisition tables have been designed: 1) for knowledge codification and extraction; 2) for knowledge validation; and, 3) for knowledge representation and system implementation. They are:

Table 1: Spatial Properties of Geographic Phenomena

Table 2: Measurement Scales of Geographic Data

Table 3: Possible Relationships between Items

Table 4: Constraints in Relation to Map Objectives

Table 5: Map Type Categories

Table 6: Comparison between Similar Map Types

Table 7: Suitability of Map Types for Given Data

Table 8: Possibilities of Combination of Different Map Types

Table 9: Visual Dimension Categories

Table 10: Visual Perception Properties

Table 11: Map Symbol Types

Table 12 Map Type vs. Symbol Type

Table 13: Visual Variable vs. Symbol Type

Table 14: The Data Properties of the Items in NAIS

There may be overlaps between the tables, but they are designed to describe the principles and methodologies used in map design from different perspectives. These tables include the knowledge introduced in chapter 2, as well as the knowledge which will be discussed in the rest of this chapter.

5.4 Knowledge on Data Properties

As shown in figure 5.3, data-driven map design is one of the main characteristics of TMDAS, therefore, data related information is very important for the reasoning procedure. Data properties related to map design were discussed in chapter 2, but, two issues need further study: spatial distribution and relationships between variables.

Spatial distribution

As described in chapter 2, spatial distribution includes both the continuous and the spatial dimension of geographic phenomena¹. This is a hierarchical description, and for cartographic application convenience, it can be further divided as follows (the importance of this division will be seen in the discussion of map type categories in 5.5.2) :

Discrete phenomena:

Point feature: A feature with zero dimension;

Line feature: A feature with one dimension;

Discrete feature²: A special type of point feature. The features exist with a high density, so that the map can only portray their distribution rather than each of the individuals (e.g., the number of people (individuals) in a region).

¹ The spatial dimension used in this thesis refers to the form of the phenomena in a generalized cartographic model, NOT in nature. For detail, refer to 2.2.1 of this thesis.

² In its narrow meaning, discrete phenomena often refers to the feature type defined here only to distinguish this type of feature from area features, because point feature and line feature are self-defined clearly.

Continuous phenomena:

Area feature with partial coverage: A feature with two dimensions, which extends unbroken and without interruptions in certain areas, but does not cover the whole region;

Area feature with total coverage: A feature with two dimensions, which extends unbroken and without interruptions within the whole region;

Volume feature: A feature which covers the whole mapping area with three-dimensional characteristics by which the values are continuously and gradually changed (which means that the interpolation can be carried out between any two known points).

These are the special distributions used in the prototype of TMDAS.

Relationships between variables

The relationships between variables in a data set is an important issue in map design. This problem was discussed earlier, in chapter 2, with three different maps (cf. figure 2.1 A,B,C), showing the significant effects of this factor on map design. But, as stated there, in conventional mapping, the knowledge of these relationships just relies on an understanding of the specific mapping objects, hence, the relationships are seldom discussed in a generic way. Further studies were carried out by the author, based on the author's own experience, and through the reading of maps and atlases, in particular, the Fifth Edition of the National Atlas of Canada. As a result, five generic relationships are abstracted and used in TMDAS; these are shown below:

Consist_of: $\text{consist_of}(A,B)$ is defined when A and B are applied to the same set of geographic phenomena. they have the same measurement scale,

and B is a subset (or a portion) of A (e.g. *Consist_of* (total_industry, heavy_industry)).

Part_of: *part_of* (A,B) is defined when A and B are applied to the same set of geographic phenomena, they have the same measurement scale, and A is a subset of B (e.g. *Part_of* (heavy_industry, total_industry)).

Neighbour_of: *neighbour_of* (A,B) is defined when A and B are applied to the same set of geographic phenomena, they have the same measurement scale, and A and B are both subsets of an item C (e.g. *Neighbour_of* (heavy_industry, light_industry)).

Same_location: *same_location* (A,B) is defined when A and B are applied to the same set of geographic phenomena, but they have different measurement scales to describe the phenomena from different aspects (e.g. *Same_location* (population_ottawa, language_ottawa)).

Related_to: *related_to* (A,B) is defined when A and B are applied to different geographic phenomena, but they are related to each other for mapping purposes (e.g. *Related_to* (factory, transportation)).

No_relation: *no_relation* (A,B) is defined when A and B do not serve the same mapping purposes.

Based on these definitions, it can then be defined that the relationships between variable 1 and 2 in the three maps in figure 2.1 are:

consist_of (C,B) in figure 2.1.A

same_location(A,B) in figure 2.1.B

related_to(A,B) in figure 2.1.C

In general, the first three relationships involve the use of compound symbols. The fourth deals with the use of "meaningful combinations" of visual variables, and the fifth implies that another map layer should be used. It is obvious that if no-relation is defined, the two variables should not be combined in one map.

As stated previously, human knowledge on data relationships is rather difficult to articulate, due to its inherently fuzzy and inexact characteristics. To represent this kind of human knowledge explicitly in formal knowledge is very difficult. The proposed scheme may be unfamiliar to both cartographers and non-cartographers, and may need to be improved or verified following feedback from users. This scheme has been implemented in the prototype of TMDAS. The technique used to help the user in defining the relationship and checking invalid replies will be introduced in chapter 6.

5.5 The Map Type Categories

Map type is a term related to classification methods by which one type of map can be distinguished from others. There are different criteria for map type classifications according to research purposes. For example, map types can be grouped according to the dominant subject matter such as land cover maps and road maps, or according to the map scale such as large scale maps and small scale maps. But these types of classifications do not meet the requirements of map design. The most important aspect of a map, from the design perspective, is its inherent characteristics related to graphic representation. Although there appears to be little classification and organization applied to the various types of representations (Forrest, 1990), map type has always been defined explicitly and in much more detail in almost all expert systems developed for assisting map design. The taxonomies used differ from system to system (refer to Laurema, et al., 1991 and Wang, 1992). In TMDAS, the map type category plays an important role. There are two modules, map type selection and map type combination, which are based on map types.

Therefore, a careful study of map types to find an appropriate classification scheme to meet the requirement of the system is a very important issue in knowledge acquisition.

5.5.1 Symbol-Driven vs. Data-Driven

It can be easily seen that to classify map types from a graphic representation perspective, the symbol type should be used, because the symbol type is directly related to the graphics. Actually, cartographers do name maps according to the type of symbols used. Robinson (1984) classified map symbolization into four classes: mapping with point symbols, mapping with line symbols, mapping with area symbols, and portraying the land-surface form. This method can be called symbol-driven and may be helpful for human cartographers to learn and do map design, but it may not apply to expert systems. This is illustrated in figure 5.4 below:

Figure 5.4 State Space of Map Design

As introduced previously, there are two kinds of states in a map design system. One is the start state, containing known facts, namely the acquired data properties. The other is the goal state, containing the solutions, namely the recommended map properties. Reasoning

Map A:
Population
of cities

Map B:
Population
of regions

Map C:
Annual rainfall
at stations

Figure 5.5 Three "Point Symbol" Maps

is used to find a successful path from the start state to the goal state, based on the known facts. Because the symbol type to be used is one of the map properties in the goal state, the goal the reasoning must find, symbol type can not be used alone to drive the reasoning. The connection to map type from the other direction, namely, data properties in the start state, needs to be identified. This method can be termed data-driven.

Before introducing the data-driven approach, an examination of the three simple maps shown in figure 5.5 will reveal the similarities and differences between the maps from both data-driven and symbol-driven perspectives. This will illustrate the significant differences between these two approaches. Figure 5.1.A shows the population of cities, figure 5.1.B the population of regions, and figure 5.1.C the annual rainfall recorded at climate stations. In the symbol-driven approach, all three maps will be called point symbol maps, because they all have employed point symbols. However, there are certain inherent differences between them. Map A is meant to show data located at specific locations, and basic questions for this map could be "what is the population of city X, and where is X located". Map B is meant to represent regional data. A common question for this map could be "what is the population in region X". Map C shows data at climate stations. A most common question for this map could be "how does the amount of rainfall change over the mapping area". Some comparison of these maps can be carried out. First, in map B, geographic units are important. There is one symbol per unit, and the position of the symbol is not important. These characteristics can not be applied to maps A and C. Second, although both map A and C represent phenomena at specific points, the points in C are sample points used to represent continuous volume phenomena, so interpolation can be carried out from map C but not A. Third, if for some reason, these map types need to be changed, the possible alternatives are different in each case. For map B, the alternatives can be 1) a dot map, or, 2) a choropleth map, after replacing the data on the number of individuals in the population (absolute quantitative) to population density (derived quantitative data). For map C, the alternative can be an isoline map (if there are enough

2

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

A resolution test chart featuring several groups of horizontal and vertical lines of varying thicknesses. Each group is accompanied by a numerical value indicating the resolution level. The values include 1.0, 1.1, 1.25, 1.4, 1.6, 1.8, 2.0, 2.2, 2.5, 2.8, 3.2, 3.6, 4.0, 4.5, 5.0, 5.6, and 6.3.

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

points to construct isolines). And there are no easy alternatives for map A. The inherent data properties makes each map different from the other two. The data-driven classification method, therefore, can divide the three maps into different categories.

As discussed previously, in data properties, two components can be differentiated. One is spatial distribution. The other is measurement scale. The latter is closely related to the visual variables to be used. The former is linked to map types. However, there are also some inter-relationships between the two components. For example, volume phenomena is a type of spatial distribution, but, by definition (refer to the definition given before), it must be portrayed as quantitative data.

Another issue which can cause confusion is the two concepts of the spatial dimension of geographic phenomena and the geometric dimension of the symbol. There is no definite match between the two. Robinson mentioned when introducing point symbols that "unfortunately not all place data are mapped with point symbols. If they were, the cartographer's symbolization problem would be easier. Likewise point symbols can map other than place data" (Robinson, et al., 1984: 285-6). Although sometimes they are matched, for example, point symbols are used for point features, it is considered to be essential to distinguish between these two at the concept level.

The data-driven approach, and the differences between the data-driven and symbol-driven approaches were introduced above. The data-driven approach presented in this thesis was inspired by the text book of the Design of Thematic Map (Zhang and Huang, 1980, in Chinese) where the map types are introduced and grouped in this manner, although the data-driven approach is not explicitly mentioned.

Although the data-driven method is important in map type classification, it does not mean that the symbol type used should not be considered at all. In fact, as an important intermedium, as defined in the concept design, map types should be connected somehow

with both states, the start state and the goal state. In the following, the TMDAS hierarchical architecture of the map type classification scheme, in which map types are connected with both data properties and map properties, is introduced.

Figure 5.6 A Hierarchical Architecture of Map Type Categories

5.5.2 Hierarchical Architecture of Map Type Categories

Figure 5.6 is a hierarchical architecture of map type categories designed for TMDAS. As shown in the figure, there are two levels to define the map categories. The first is data property, which relates the map type to the data. The second is symbol type, which relates the map type to the symbol. Therefore, at the bottom level, each map type includes the type of data and the kind of symbol – these are the distinct properties of map type that TMDAS requires. Each map type together with example maps will now be introduced.

5.5.2.1 Mapping Point Features

Point Map: a map which represents the geographic phenomena which at a given scale can only be represented in point form (zero-dimension), by using point symbols (refer to Robinson, et al., 1984: 285). The point symbol's location represents the location of that point feature, and can be used to represent data at any measurement scale. Figure 5.7 is an example of a point map.

Figure 5.7 Example of a Point Map

5.5.2.2 Mapping Line Features

Line Map: a map which represents the geographic phenomena which at a given scale can only be represented in line form (one-dimension), by using line symbols (refer to

Robinson, et al., 1984: 312). The lines are normally located at exact positions, and are often used to represent linear features at nominal or ordinal scale. Figure 5.8 is an example of a line map.

Figure 5.8 Example of a Line Map

Flow Map: "maps that show linear movement between places are called flow maps, and because of this quality, are sometimes referred to as dynamic maps" (Dent, 1991: 259). The distinct characteristic of this map is to show the movement of phenomena. The position of the line is normally less accurate than in the line map, and is sometimes even generalized to show the connection between places and the direction of the movement. Flow maps can represent data at any measurement scale, but this type of map design is normally very complex, and is thus not included in the prototype of TMDAS. Figure 5.9 is an example of a flow map.

Figure 5.9 Example of a Flow Map

5.5.2.3 Mapping Discrete Features

Dot Map: "mapping discrete geographical phenomena can be accomplished by dot mapping, also called aerial frequency mapping" (Dent, 1991: 183). From the point of view of the symbol used, it is a point symbol map, because dots are a repetition of point symbols. But, as Robinson pointed out, "conceptually the dot map is more complex than the more simplified point symbol map" (Robinson, et al., 1984: 300). The term "areal frequency mapping" can be used to describe the property of this type of map. In a dot map, each dot is assigned a specific value and the dot's placement is an attempt to show data distribution. The ordinal perception "...is probably the most important function to be performed by the dot map" (Robinson, et al., 1984: 300). Figure 5.10 is an example of a dot map.

Figure 5.10 Example of a Dot Map

5.5.2.4 Mapping Area Features with Partial Coverage

Area Map: a map which represents the geographic phenomena which are in area form (two-dimension) but do not continuously cover the region, by using area symbols. There is normally no distinction between category maps and area maps, because they are both used to represent area data at nominal measurement scales. But "...some minor problems can be involved. Frequently, nominal categories are not geographically exclusive; that is, two or more categories often occur in the same area... if the cartographer does not want to generalize sufficiently to remove the mixture, symbolization that somehow represents the mixture or overlap must be employed" (Robinson, et al., 1984: 340). In other words, it is important to know whether the area feature covers the whole area or not, with or without overlaps. This will affect both the visual variable selection and the estimation of map

density. Therefore, they are separated as two map types here. Figure 5.11 is an example of an area map.

Figure 5.11 Example of an Area Map

5.5.2.5 Mapping Area Features with Total Coverage

Category Map: a map which represents the geographic phenomena which are in area form (two-dimension) and cover the whole region, by using area symbols. Category maps are always used to represent nominal scale data such as land-use or administrative divisions. In this type of map area symbols will usually cover the whole map. Figure 5.12 is an example of a category map.

Figure 5.12 Example of a Category Map

5.5.2.6 Mapping Volume Features

Isoline Map: a map which "...involves mapping a real or conceptual three-dimensional geographic volume with quantitative line symbols" (Dent, 1991: 216). Shading between isolines, named layer-tint, is often applied together in this type of maps. Robinson (1984) divided isarithmic mapping into two types. One is isoline mapping to be used to map natural data with values that can occur at points, such as annual rainfall. The second is isopleth mapping, which maps statistical surfaces, with values that can not occur at points, such as persons per square kilometer (for detail, see Robinson, et al., 1984: 320-21). The data to be mapped by isopleth mapping is normally social and economic data. Although there is no distinction on the use of cartographic symbols for these two types of maps, the concepts and the data to be represented are quite different. Unlike natural volume data, social and economic data usually do not change continuously nor gradually, so, isopleth

maps should be produced with caution. Therefore, only isoline maps are included in the prototype of TMDAS. Figure 5.13 is an example of an isoline map.

Figure 5.13 Example of an Isoline Map

Located Graphics Map: a map which represents geographic phenomena which are in volume form (three-dimension), especially data which changes periodically, by using special point symbols (e.g., bar-chart). These point symbols are often used and located at sample points (e.g., climate stations) to show the distribution and change of the phenomena over the region. Figure 5.14 is an example of located graphic map.

Figure 5.14 Example of a Located Graphics Map

5.5.2.7 Mapping Aggregated Data

Choropleth Map: a map which represents data from a derived quantitative measurement scale and aggregated by geographic area units, by using area symbols. A choropleth map is a type of statistical map which has been widely used to represent social and economic data. The geographic area units are census divisions or other political divisions. Two issues should be emphasized in producing choropleth maps. First, "geographical phenomena that are continuous in nature should not be mapped by the choropleth technique since their distributions are not controlled by political or administrative subdivisions" (Dent, 1990: 148). Second, absolute quantitative values (total values) should not be used in choropleth mapping, because the varying size of the units and their mapped values will alter the impression of the distribution. This is an area where novice cartographers can

often make mistakes. The choropleth map is also the type of map in which area symbols usually cover the whole region. Figure 5.15 is an example of a choropleth map.

Figure 5.15 Example of a Choropleth Map

Area-based Point Map: a map which represents aggregated data from an absolute quantitative measurement scale and aggregated by geographic area units, by using point symbols. The area-based point map is also a type of statistical map in wide use. The measurement scale is an important issue in producing area-based point maps. In contrast to making a choropleth map, absolute quantitative data must be used here. Actually area-based point maps are often used together or interchanged with choropleth maps in practice, according to the measurement scales of the data. Figure 5.16 is an example of an area-based point map.

Figure 5.16 Example of an Area Based Point Map

5.5.2.8 Summary of Map Type Categories

The map types introduced above are all commonly used graphic representations in map design practices. What is unfamiliar may be the way of grouping them, or the way of describing them, or, in particular, the terms used for them. As it has been designed, this classification scheme has three main characteristics: 1) it is based on the data properties, which connect the automatic reasoning with the start state; 2) each map type has been explicitly associated with symbol types, which connect the automatic reasoning with the solution state; 3) because each map type carries the information of the data properties, it is relatively easy to select an alternative for a map type when it is needed based on these data properties. In other words, it is possible, based on these map type categories, to establish a map type change network to indicate which two map types can be used alternatively and

under what conditions. Map type change may be needed in the design of multi-variable mapping.

The naming of map types is the most difficult aspect in this classification scheme. For the maps which already have widely accepted names, such as choropleth maps, these names were used. Some names had to be produced. All the terms used here are for convenience of description in the thesis and in TMDAS.

Map types are not part of the final recommendation which consists of symbol types and visual dimensions only. In fact, a map type is not the basic information needed for map design. It is just an intermedium and is important to the developer, but not to the end-user. The application of these map types and their usage will be introduced in chapter 6.

5.6 Possibilities of Map Type Combination

Map type categories and map types have been introduced. In this section, the issue of map type combination will be discussed. For multi-variable mapping, there are usually more than one map layer involved. A check for map layer combination, therefore, is needed to answer: 1) Can these single layers be put together to form a multi-layer map which meets the requirements of map legibility? 2) If so, are there any restrictions for any of the single layers? 3) If not, are there any alternatives? When comparing each map type with all other map types in turn, it can be concluded that there can be three cases for combinations of any two map types:

Permitted (P): the combination of the two layers is permitted without restrictions. For example, a point map and a line map can be easily combined.

Permitted With Restrictions (PWR): the combination of the two layers is permitted, but with some restrictions. For example, when two area maps

are combined, visual variable *hue* can only be applied to one of them, because the two areas may have overlaps, and the combination of the two colours may form a new colour which causes confusion.

Not Recommended (NR): the combination of the two layers should not be used.

There are two cases which can cause the third type of combination. First, the two map types may represent two different data which do not serve the same map objectives. For example, an isoline map (mapping a natural feature) and a choropleth map (mapping social and economic features) are seldom used together. In this case, this type of map should not be permitted. Second, it may be difficult to combine the two map types cartographically, that is, the legibility of the map can not be retained, or the combination may create a misleading map. For example, the combination of a category map and a choropleth map can be quite disturbing. If this is the case, it is necessary to further check for alternatives for the two map types with or without additional data processing.

Each map type has to be checked against all others to define the case of each combination: permitted, permitted with restrictions, or not recommended. If the combination is of the second type, the restrictions must be clearly defined for the rest of the design. In the third case, it is necessary to define alternatives, or to give advice with explanations.

Another issue of map type combination is the control of the total capacity of the map. The map type combination check mentioned above indicates whether any *two* layers in the map can be used together. It is also still necessary to check whether all the layers can be assembled on one map sheet. In TMDAS, the following factors are considered: map purpose related factors, the number of map layers, and the map types applied to each layer. In this way, the control of the total capacity can be accomplished with flexibility. For example, two choropleth layers would be the maximum for a paper-based map (often

referred as a bi-variable map by cartographers), while the combination of one point layer, one line layer, and one area layer could still be handled. But what must be emphasized is that although the system takes a large number of factors into consideration, it still can not work as flexibly as a human cartographer. In conventional cartography, there are always exceptions to rules, for special reasons. These special cases cannot be included in the system. Moreover, an important factor, the direct vision of the graphic form, is missing in the system. As a result, the rules developed for the system are often conservative.

5.7 Use of Visual Dimensions

Visual dimensions and their perception properties have been discussed in detail in chapter 2. The key point of using visual dimensions in mapping is that the organization level of the visual variables should match the organization level of the data (refer to chapter 2), otherwise, a misleading map can be created. For example, let us compare the two maps in figure 5.17. If the purpose of the map is to show data defined in legend A, the population of cities, map A is correct and map B is wrong. The reason is that the visual variable used in A is size which has quantitative perceptions, while the visual variable used in B is form which does not offer quantitative perceptions. However, if the purpose of the map is to show data defined in legend B, the dominant language in cities, B is then right and A is

Figure 5.17 An Example of the Organization Level of Visual Variables

wrong for a similar reason. Therefore, the selection of visual variables should be based on the measurement scales of the data. In addition, the symbol type used and the medium of the map should also be considered. The following section discusses these issues in detail.

5.7.1 Mapping Nominal Data

Based on the table shown in figure 2.2, in general, there are three visual variables used to map nominal data, namely: hue, form and orientation. Their selective properties, which are the important properties for nominal data mapping, are not the same when applied to points, lines, or area features, as shown in figure 5.18. There are some practical limitations in cartographic design (see Robinson, et al., 1984). Three visual variables can be applied to point symbols. However, form and orientation are often not used for line symbols, and orientation is not recommended for area symbols if used alone.

Visual variable	Symbol type				
	Point		Line		Area
Hue
Form
Orientation

Figure 5.18 Visual Variables for Mapping Nominal Data
(based on Robinson et al., 1984)

Cartographers often call the form a pattern when it is applied to area symbols. In fact, the pattern is a regular repetition of a symbol with a specific form. The important point in designing patterns is that the overall value perception should be kept constant.

5.7.2 Mapping Ordinal Data

Ordinal perception should be maintained when mapping ordinal data. For this purpose, visual variables of value, spacing, and size are preferable. But considering the legibility of the map, as shown in figure 5.19, spacing for point and line symbols can only be used under certain conditions (Robinson, et al., 1984) such as the minimum size of the symbols, while size for area symbols is seldom used. Therefore, these cases are not included in the system.

Visual variable	Symbol type				
	Point		Line		Area
Spacing
Value

Figure 5.19 Visual Variables for Mapping Ordinal Data
(based on Robinson et al., 1984)

Value is considered as the most suitable visual variable for ordinal data, especially for screen based maps. Spacing is actually another way of presenting differences in value, and it is often used in hard copy maps produced by pen plotters. Size has to be used carefully here, because size also has a quantitative perception. Size is frequently used to portray line

features for ordinal perception, but it is used differently for proportional symbols, where a proportional relationship exists between data and symbol. For ordinal data representation, an equal difference should be kept between adjacent symbols, to express the characteristics of the ordinal data.

In thematic mapping, there is sometimes a need to represent data at a level lower than the level of data measurement scale. For example, to represent ordinal perception based on quantitative data, that is, to group quantitative data into ordinal groups. It should be emphasized that after this grouping, the data property to be represented is no longer quantitative. Therefore, all the rules for representing ordinal data must be followed. When size is used to represent ordinal data in TMDAS, it is termed "size-nonp", which means that size is used in a non-proportional way, to distinguish it from a proportional use.

5.7.3 Mapping Absolute Quantitative Data

Based on the table in figure 2.2, size is the only visual variable which has quantitative perception. The most often used graphic representation in size is the so called proportional point symbol. When size is applied to a line symbol, it is often referred to as width. It is frequently used in flow maps. To apply size to an area symbol, there is a type of map, named value-by-area (see Dent, 1990). However, this map type is not very popular, due to its special construction method, so it is not included in TMDAS. In practice, the most common method to represent area quantitative data is either the area-based point symbol map or the choropleth map depending on the type of quantitative data (as discussed earlier). When size is applied to compound symbols such as pie-charts or bar-charts, size may be referred to as angle, height, etc.

5.7.4 Mapping Derived Quantitative Data

Derived quantitative data can be ratios involving area or ratios independent of area. To separate quantitative data and absolute derived data is very important for mapping based on

geographic area units. As introduced previously, only derived quantitative data can be mapped with choropleth maps using area symbols. But what visual variable should be applied to a choropleth map? The solution recommended by Bertin (1983), shown in figure 5.20, is called *graduated sizes in a regular pattern*, because he insists that "above all, it is the most rigorous of all graphic solutions for quantitative problems" (Bertin, 1983: 127). But in cartography, the accepted method is to apply visual variable values to classified data. There are two main disadvantages to this conventional method: first, visual variable values do not have a quantitative perception; second, the graphic impression can be changed by different classifications. However, Bertin's method also seems to have practical restrictions. It remains a topic for research in attempting to improve the representation of choropleth maps. In the development of TMDAS, the conventional method, the use of value, is adopted. When there is a need to increase the length of the scale, or to meet a special requirement, the combination of two variables, value and hue, can be applied.

Figure 5.20 Graduated Sizes in a Regular Pattern
(based on Bertin, 1983)

5.8 Visual Constants and Further Aspects in Design

There are two ways, as discussed in chapter 2, to apply visual dimensions: variation and uniform. The former is the visual variable which has previously been discussed. The latter

is the visual constant which does not change with variations in data. The number of visual constants involved can be one, or more, depending on the symbol types and visual variables selected. The visual constant may not be as important as the visual variables, but defining the visual constants is also part of the design, especially for automated mapping. In some of the cases, it is better to be aware that some visual dimensions should be kept constant. A common mistake occurs in the production of category maps with hue variables: because values used for different hues are not constant, there is an ordinal perception which this type of map should not have. Therefore, visual constant is also included in TMDAS.

There are additional aspects which should be mentioned in reference to the framework of concept design given in figure 4.3. There is another module in the framework, the actual graphic design module, to complete the map design. For instance, if a dot map is recommended, what then should be the value assigned to a dot in order to represent the distribution properly? If a choropleth map is to be produced, how many classes are needed and what should the criteria for classification be? What is the proper range of the sizes for proportional point symbols? To solve these issues, a visual justification from the graphics produced is often needed through interactive work with the graphics. Visual justification is very difficult to accomplish in a fully automated environment, because the procedure has not been simulated. Pattern recognition technology may be worth consideration. But in the prototype of TMDAS, this module is not included, hence, the knowledge associated with this aspect is not discussed.

5.9 Knowledge Extraction from NAIS Data Set

NAIS is considered as a specific data set in TMDAS for three reasons. First, as the National Atlas Information System, it provides "the background information for 44 different thematic realms, including physical themes (geology, climatology, forestry, etc.), human themes (population, languages, religion, etc.), economic themes (agriculture,

communication, employment, etc.), and historical themes (settlement, ethonography, culture)" (Richardson and Muller, 1991: 141). Therefore, it can be used as a rich thematic data bank for mapping purposes. Second, because of this potential, a codified NAIS knowledge base will help the general user in becoming familiar with NAIS. Third, this specific data base is also designed as a test on the flexibility of the system design. It proves that a theme-oriented sub-module can easily be added to TMDAS simply by adding a stand-alone knowledge base. The knowledge in the NAIS data set is extracted, based on the reading of the National Atlas of Canada Fifth Edition, and related documents. Because the knowledge base consists of the data properties of NAIS data, the user can select the data to be mapped from the list, then the data properties will be matched automatically. Therefore, should any data set with frequent access be required, a special knowledge base could be developed and added in the same way. At the same time, the cartographic function and data property oriented design will guarantee that the system will work for any data which are not included in those specific knowledge bases.

5.10 Meta-Knowledge Extraction

Meta-knowledge is the knowledge used in the reasoning to control the system and to make decisions on what action to perform in specific cases. Meta-knowledge is important for establishing a user friendly interface, and for increasing system efficiency. Some important aspects are briefly discussed below, and their implementation will be described in chapter 6.

Checking valid response: when a human expert is consulted, for each question that the expert is asked, there is a chance to get an unexpected answer. The expert needs first to check whether the answer is reasonable, and, if not, the expert may know that some extra information is required to guide the conversation in a correct way. The same mechanism should also be available in an expert system. There can be two kinds of invalid responses. One is technically invalid. For example, the selection is outside of the range of the given

multiple choices. This kind of control can easily be added to the system. The other is logically invalid. For example, the relationship between a point feature set and a line feature set is defined as the "same-location" (refer to the definition given before). The knowledge to ensure against these errors should be extracted carefully to avoid unacceptable system performance.

Guiding the control flow: during the reasoning procedure, there is always more than one factor being considered. The result of the reasoning for each factor can influence the others as well as the entire control flow. Sometimes, it may be necessary to go back to recheck certain things, or a point may be reached which makes the rest of the reasoning unnecessary. This strategic knowledge is very important for system implementation. The extraction of this knowledge relies on an understanding of the problem domain and of the entire structure of the knowledge. Strategy knowledge needs to be checked and updated with prototype implementation.

Giving explanations: in case users have difficulties using the system, there should always be helpful information, just as when dealing with a human expert. When the system asks a question, users may have difficulty understanding the question, or they may be puzzled by why that question is being asked. When the system gives a result, the user may want to know how that solution has been reached, in order to justify its reliability. Part of these facilities can be accomplished by the inference engine of the system (will be discussed in chapter 6), and part of them are related to knowledge, and, need to be extracted and stored in the knowledge base.

5.11 Knowledge Validation

Knowledge validation is an important issue in expert system development, because the level at which the system can give expert level advice depends on the quality of the knowledge. Yet, knowledge validation checking is difficult, because after the knowledge

has been decomposed and structured into a formal knowledge representation, the domain expert may not feel familiar with the knowledge any more, since domain experts are not usually knowledge engineers. This makes it very difficult to recognize what is correct decomposition, and what is mis-matched representation. Therefore, prototype development is considered essential for expert system development. Case tests of the prototype are the best way to check knowledge validation. Generally, knowledge base updating based on the feedback from case tests must be carried out. This procedure must be repeated until the result is satisfactory. In the literature, this whole process is sometimes included in the stage of knowledge acquisition (refer to Gruber, 1989: 129), because case tests may indicate that another knowledge acquisition stage is needed. Prototyping and case tests are the main methods used to check the knowledge validation in TMDAS, although a few expert interviews were also conducted. Some of the case tests and analysis will be introduced in chapter 7.

5.12 Summary

Selected issues involved in knowledge acquisition in TMDAS have been discussed in this chapter from two perspectives. One is the knowledge engineer's perspective. The knowledge, such as the state space representation of the problem, problem decomposition, knowledge modelling, and inference controlling, has been introduced. The other is the domain expert's perspective. Knowledge involved in map design from different aspects has also been further developed. It should be emphasized that all the knowledge introduced in both chapter 2 and this chapter form the knowledge base of TMDAS. Chapter 2 concentrates on knowledge codification as well as extraction from some authoritative text books to establish the knowledge frame for TMDAS. Therefore, it provided a background for TMDAS. But the knowledge acquired in chapter 2 is not sufficient to support TMDAS. So, this chapter can be considered as the further development of the knowledge. Together they offer a solid body of knowledge for the implementation of TMDAS.

The knowledge introduced in this chapter is one of the main contributions of this project, but because there are so little experimentation done, there can be some problems with the use of this knowledge. However, what is important here is that this project indicates that certain aspects in map design and in cartography are still in need of much research. Therefore, the real intention is to draw attention to these aspects rather than to introduce solutions. The development of cartographic theory and concepts will be the main stimulator for the development on expert system in cartography, and vice versa.

6. SYSTEM IMPLEMENTATION

6.1 Introduction

In expert system development, prototype implementation is an important procedure, because, unlike in conventional systems, the proper representation method normally can not be found until it is reached. The stages in expert system development are often not as clear as those in conventional system development. Prototyping is actually a procedure of using the computer to help explore a problem's dimensions and to develop understanding of the problem. Expert system programming is often called exploratory programming. Luger and Stubblefield stated, "Like most AI programming, expert system development requires a nontraditional development cycle based on early prototyping and incremental revision of the code. This exploratory programming methodology is complicated by the interaction between the knowledge engineer and the domain expert" (Luger and Stubblefield, 1989: 299).

Figure 6.1 is a illustration of this general exploratory programming procedure. The three dashed lines (added by the author) should be noted; the dashed lines can sometimes form an infinite loop, and the project can fail.

In this chapter, the development of TMDAS is introduced through the themes of the selection of a system environment, the design of system performance, the development of the expert system shell, the construction of the knowledge base, and the knowledge formalization.

6.2 Environment for System Development

In selecting a system configuration for TMDAS development, the following factors were considered. The first is capability. The system should meet all the requirements of the

Figure 6.1 Exploratory Programming (from Luger, 1992: 393-6-33)

concept design, be able to handle a knowledge base of a given size, have good graphic functions, and be easily extended to access a GIS/EA database. The second is popularity. The system platform should be inexpensive and popular. The third is availability. Because the research is undertaken as part of a thesis, there is little financial support for both hardware and software investigation. The fourth is familiarity. Time is not available for intensive market investigation.

6.2.1 Selection of Platform

Based on these considerations, an IBM-PC compatible is selected as the system platform. PCs have been rapidly developing in recent years. From the 286, and 386, to the 486, the

performance/cost ratio has increased. Because many PC-based GIS and CAC (Computer Assisted Cartography) software packages have been used in the automated mapping environment, the use of TMDAS will not require more powerful hardware. In order to make TMDAS usable in as many cases as possible, only a basic configuration is required. The simplest configuration is:

IBM PC Compatible

640K RAM

20M Hard Disk

1 Floppy Disk Drive

Colour Graphic Monitor

VGA Graphic Card (640*480).

6.2.2 Selection of Programming Language

The software environment for expert system development is normally different from the environment for conventional programming. For prototype implementation, available expert systems shells or AI programming languages are often used. The former is easier to use, because it has the inference engine, the user interface, and the explanation facilities built-in. But the main disadvantage of expert systems shells is their inflexibility. Because shells are often the by-product of earlier successful applications, they are often only suitable to the domains related to these applications. It is thus necessary to separate the domain knowledge from the reasoning algorithms. Therefore, it was decided to develop an expert system shell based on the requirements of TMDAS, and then to construct the knowledge base with domain related knowledge only. In the implementation of TMDAS, two languages were used, Turbo Prolog, and C.

6.2.2.1 Turbo Prolog

Turbo Prolog, from Borland International, is one of the most popular microcomputer Prolog implementations. It has distinctive merits as well as short-comings. The main disadvantage of this language is that all data types must be declared in advance and the predicates with the same name must be grouped together (it has been argued that these disadvantages were against the principles of Prolog as a declarative language. It is not recommended to learn the Prolog concept from Turbo Prolog). However, Turbo Prolog also has advantages. The first is that when a program written in Turbo Prolog is compiled, it can run much faster than interpreted ones. The second is its fast program execution (because of those restrictions mentioned above): it is two or three times faster than compiled Arity Prolog (Covington, et al., 1988: 458). Moreover, it is easy to access the hardware, operating system or other programming languages with Turbo Prolog. These flexibilities, the high execution speed, as well as availability were the main factors in the selection of Turbo Prolog. Some basic concepts and technical issues of Turbo Prolog used in this project are briefly introduced below.

Program structure

The Turbo Prolog program must begin with a set of *type declarations*. A typical program has the following structure:

```
DOMAINS
    data type declarations
DATABASE
    dynamic database predicate declarations
PREDICATES
    predicate declarations
CLAUSES
    predicate definition
```


GOAL

starting query

For example, a program to append two lists can be written in the following manner:

DOMAINS

list = symbol *

PREDICATES

append (list, list, list)

CLAUSES

append ([], X, X).

append ([Head|Tail], Y, [Head|Z]) :- append (Tail, Y, Z).

GOAL

append ([a, b], [c, d], X),

write ("The new list is "),

write (X).

After running this small program, the printed result would be: " The new list is a, b, c, d".

Annotations and data types

Turbo Prolog is based on the first-order predicate, but uses some different annotations as shown below:

Predicate calculus	Turbo Prolog
$\wedge$ (and)	.
$\rightarrow$ (implies)	:-
$\neg$ (not)	not

There is no $\vee$ (or) operator, instead, this expression must be represented by two entries.

Variables in Turbo Prolog are expressed with the first letter in upper-case. For example,

Colour in Turbo Prolog is a variable, while *red* is a constant. A period "." is required at the end of each predicate.

In Turbo Prolog, there are six basic data types: char, string, symbol, integer, real, and file. In addition, one of the strong points of Turbo Prolog is that it supports more complex and useful structures based on these six data types. For instance,

DOMAINS

list = symbol *

means that "list" is a user defined data type, which is a *list* of symbols (symbol is one of the basic data types). After new data types are defined, they can be used in the same manner as basic data types.

Unification

Unification is the most fundamental function used in symbolic programming. It is a technique which defines the matching process to find a solution. For example, suppose there are facts stored as:

clauses

measurement_scale (population, absolute_quantitative).

measurement_scale (landuse, nominal).

measurement_scale (road, ordinal).

and the goal is specified as:

measurement_scale (landuse, X).

To achieve this goal, Prolog would search the clauses for a match. Two terms can be matched only if they both have the same predicate, in this case, "measurement_scale". After the constant, landuse, is matched, the variable X is then unified to "nominal". So, the solution reached is that "the measurement scale of landuse is nominal". In unification, a free variable (which has not been related to any constant) can be unified with any term including another free variable. After being unified, the free variable is bound to that term,

and is called a bound variable. A constant can only be unified with itself or a free variable. The use of unification in TMDAS implementation will be discussed in later sections.

List and recursion

List is a very important and powerful data structure in AI programs. A list is an ordered sequence of zero or more terms, written between square brackets, and separated by commas, such as:

symbol_type [point, line, area].

Lists can be constructed or decomposed through unification. For instance:

Unify	With	Result
[a, b, c]	X	X = [a, b, c]
[a(b), c(X)]	[Z, c(a)]	X = a, Z = a(b)

Moreover, any list can be divided into a *head* and *tail* by the symbol "|". The head is the first element, and the tail is a list of the remaining elements (it can be empty). For example:

Unify	With	Result
[a, b, c]	[H T]	H = a, T = [b, c]

The power of the list relies on the power of recursion, a technique for list processing. *Recursion* is a program structure in which a procedure calls itself. The idea is that, to solve a problem, some action is performed to solve a smaller problem of the same type using the same procedure. The process terminates when the problem becomes so small that the procedure can solve it in one step without calling itself again (Covington, et al., 1988). Therefore, there should be two parts in a recursive procedure: calling itself to repeat a sequence of actions, and terminating the conditions that stop the procedure. Recursion can be seen in the example "append" given previously:

`append ([], X, X).` –terminating conditions (when the list to be appended is empty)

`append ([Head|Tail], Y, [Head|Z]) :- append (Tail, Y, Z).` – calling itself

Lists and recursion are very important data structures and techniques used in the development of the expert system shell for TMDAS (to be introduced later).

Dynamic data base

Turbo Prolog can define an internal data base by using a DATABASE declaration. An internal data base is composed of facts (not rules) that can be added or removed from the program at run time. The important use of this dynamic data base in TMDAS is that it can retrieve dynamic knowledge (facts derived from inference) or intermediate results and save them for later use.

The Turbo Prolog compiler requires only 384K of memory space. This makes it possible to run Turbo Prolog on most IBM compatible computers. But this is very inconvenient for large programs such as TMDAS, which require a large amount of space. The program must be broken into pieces physically, which is not always easy. Unlike conventional programs that are often used for computing and are easy to divide into steps, expert system programs involve reasoning, in which backtracking is often encountered, so it is usually difficult to break the entire procedure into separate steps. A dynamic data base is thus very useful in handling this problem.

6.2.2.2 The C Programming Language

C is one of the most widely used procedural languages in programming due to its advantages of a small code size, fast execution, more flexible data structure, and the ability to access hardware resources. In comparison to Turbo Prolog, C is better at both handling large locational data files and dealing with graphics. Therefore, C is used for these two

aspects in the implementation of TMDAS, mainly in the development of the user interface and the on-line map library.

The Borland C/C++ compiler is used in this project. The compiler package provides full support of the C/C++ language standard and includes a graphic library which has many functions.

6.3 The Design of System Performance

Before discussing technical issues of the implementation of TMDAS, the design of the system performance, as shown in figure 6.2, must be introduced. This design is the frame used in the implementation.

Figure 6.2 System Performance Design

In the diagram, J.shell, the expert system shell, and the TMDAS knowledge base form the body of the system; they will be discussed in detail later. The three boxes at the top of the diagram represent three information query modes that the user can select, based on the data to be mapped. The system-user interface on the left is a bi-way communication mechanism through which both the system and the user can acquire information from each other. When the system asks questions, the user can either answer back, or ask either for the further explanations of the question, or the reason for the question. The two windows on the right are designed for this purpose. If the user answers back, the system will check if the answer is valid. After obtaining all the needed information, the system will begin the reasoning process. Generally speaking, the final recommendation will be given in about 10 seconds (for PC 386, 33MHz.) in the form shown in figure 6.3.

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Oil_pipelines 1	line/ line	hue	value, size
2	Pump_stations 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue
3	Oil_fields 1	area/ area	hue form	value size, value, spacing

Figure 6.3 An Example of a Recommendation from TMDAS

The final recommendation includes the number of layers for the map, the number of variables and the variable names in each layer, the map types selected for each layer, and most importantly, the symbol types, visual variables and visual constants selected for each

layer. The user will then have four choices: 1) SAVE the recommendation as an ASCII file which can be printed; 2) VIEW the example maps corresponding to the recommendation; 3) ask HOW was the recommendation reached to see the solution path either through STATIC, which displays the whole solution path at once, or DYNAMIC, which displays the path node by node; or, 4) CONTINUE the process to conduct another consultation or quit the system. In the rest of this chapter, the manner in which this design was accomplished, will be discussed.

6.4 The Development of J.shell – an Expert System Shell

Prolog has a built-in inference engine which can carry out the basic pattern matching function. But it was necessary to develop an expert system shell for the following reasons. First, when using a programming language directly, it is necessary to pay attention to both the knowledge aspect of the system, and the algorithms needed to apply the knowledge. They will be highly interwoven and not easily separated (Lucas and Gaag, 1991). This causes difficulties when updating the knowledge, which is always necessary, because, expert knowledge is dynamic in nature: it is subject to change. Secondly, after the shell is developed, any knowledge base, as long as it is written according to the format defined by the shell, can be supported by the shell. Therefore, the knowledge base can easily be extended. Thirdly, it is much easier to implement the explanation facility and user interface separately in order to offer the explanation in the way an expert system should, "to qualify as true expert systems" (for detailed discussion, refer to Buittenfield and Mark, 1991). Therefore, J.shell, a simple expert system shell, was developed for these purposes.

6.4.1 The Basic Format Used in J.shell

J.shell is a rule-based shell constructed in Turbo Prolog. Each piece of knowledge in the knowledge base is called an entry, and can be represented, in one of the following formats, as predicates:

PREDICATES

fact(fact_number,fact).

rule(rule_number,then,ifs).

question(question_number,question).

explanation(explanation_number,question,explanation).

utility(utility_number,utility).

validresponse(validre_number,question,response,check_response,converted_response).

In these expressions, the symbols used before the parentheses "(" are the names of the predicates. The symbols between the parentheses and separated by commas indicate the number of elements in the predicate and their data types. In J.shell, their types are defined as follows:

fact_number, rule_number, question_number, explanation_number,

utilities_number, validre_number: INTEGER

explanation, response: SYMBOL

fact, then, question, utilities, check_response, converted_response: DATA_TYPE

ifs: DATA_LIST

DATA_TYPE and **DATA_LIST** are user defined types. **DATA_TYPE** includes all data types used in the system, and **DATA_LIST** defines a list whose elements have types defined by **DATA_TYPE**. In addition, the term "initialize" is used as the start to the reasoning. By using this shell, the knowledge base developers can concentrate on the representation of knowledge in the defined formats, and need not worry about the reasoning. Below are further explanations on the use of each formats, with simple examples.

Fact (fact(fact_number,fact)).

Fact is the simplest format to represent knowledge. It can be used to express a property or a relationship, and return a boolean value of *true* or *false*.

fact_number: the number assigned to the fact.

fact: the knowledge represented in any defined data type.

Example: fact (25, no_variable(8)).

This entry indicates that this is fact number 25, and that content is "the number of variables is 8".

Example: fact(38,layer_combine(point_symbol,line_symbol,permitted)).

This entry indicates that this is fact 38, and that its content is "a point symbol layer combination with a line_symbol layer is permitted".

Rule (rule(rule_number,then,ifs))

Rule is the main format to represent knowledge because it can extract new knowledge from known information. The format used here is actually another way of representation of the production rules introduced in chapter 3, with an easier implementation in Prolog.

rule_number: the number assigned to the rule.

then: the head of the rule which defines the conclusion of the rule

ifs: the body of the rule which defines the conditions which must be met in order to make the conclusion valid.

Example: rule(338,layer_map_type(L,area),

[layer_spatial_dimension(L,partial_area),

layer_measurement_scale(L,nominal)]).

This entry indicates that this is rule number 338, and that its content is "if the spatial dimension of the layer is 'partial_area' AND the measurement scale of the layer is 'nominal', then the map type of the layer should be an area symbol map".

(If the above rule is represented by a production rule, it will be:

```
IF      layer_spatial_dimension(L,partial_area),  
        layer_measurement_scale(L,nominal)  
THEN layer_map_type(L,area)).
```

The body of the rule ("ifs") is represented as a list. The inference engine takes each element as a subgoal and attempts to prove each in turn. Each of the subgoals can be a fact, the head ("then") of another rule, or it can be related to a question, when the information is required by the user.

Question (question(question_number,question))

Question is constructed as a simple user interface for information acquisition by the user.

Prolog annotation is directly used here.

question_number: the number assigned to the question

question: the term used for the question

Example: question(108,screen_map(X)):-

```
nl,write("The final map will be on:"),  
nl,write("    1. screen;"),  
nl,write("    2. paper;"),  
nl,write("Enter your choice: ").
```

This example shows that whenever the inference engine encounters the subgoal of "screen_map(X)", it will recognize that the information needs to be entered by the user.

Then, the following message will appear on the screen:

The final map will be on:

1. screen
2. paper

Enter your choice:

After receiving the answer from the user, the inference engine first checks if the answer is valid, then inserts the result into a dynamic database for later use to ensure that the question is not asked again.

Validresponse (validresponse(validre_number, question, response, check_response, converted_response))

"Validresponse" is designed so that knowledge base developers can define the valid response explicitly and include any restrictions in the answer.

validre_number: the number assigned to the valid response

question: the term used for the question

response: the answer given by the user

check_response: the function by which the response is checked

converted_response: the internal representation of the answer to be stored in the dynamic database

Example: validresponse(104, screen_map(X), Response,
check_chr("1",Response,"1"), screen_map(yes)).

This entry indicates that for the question "screen_map(X)", if the user answers "1", then the answer is valid, and the system should insert 'screen_map(yes)' into the dynamic database".

For questions which do not have a defined valid response, the inference engine assumes that any answer is valid, and inserts the answer directly into the dynamic database.

Explanation (explanation(explanation_number,question,explanation))

The explanation predicate is designed to supply further information about the question asked.

explanation_number: the number assigned to the rule

question: the term used for the question

explanation: the content used for explanation

Example 6: explanation(101,spatial_dimension(Variable,D),"Spatial dimension ...").

This entry indicates that for the question 'spatial_dimension(Variable, D)', display the information as 'Spatial dimension...'. This entry will be accessed when the user is requiring further information on the question, asked by the system, requiring the spatial dimension of a variable. The information will appear in the EXPLANATION WINDOWS.

Spatial dimension ...

If there is no information stored in the knowledge base, the following information will be displayed instead:

Sorry! No further information available for this question

Utilities(utilities_utilities_number,utilities))

Utilities is a predicate designed for the knowledge base to access the Turbo Prolog built-in predicates indirectly. For technical reasons, the built-in predicates cannot be used in the user defined "rule" predicate where they are frequently needed. The function of the utilities predicate is then to access the built-in predicates through the use of prolog annotations.

utilities_number: the number assigned to the rule

utilities: the term used for the utilities

Example 7: utilities(308,mynot_e(X1,X2)) :-not(X1=X2).

This entry indicates that "the term 'mynot_e(X1,X2)' access the built-in predicate 'not(X1=X2)'. Therefore, whenever there is a need to use the built-in predicate 'not(X1=X2)' in the construction of a rule for checking whether the two variables are equal, 'mynot_e(X1,X2)' can be used as a substitute.

6.4.2 The Inference Engine of J.shell

The J.shell inference engine employs backward-chaining (introduced in chapter 3) to process the knowledge stored in the knowledge base. Any knowledge base built in the representation scheme introduced above can use J.shell as the inference engine by using the following two steps:

- 1) add the following at the beginning of J.shell:

include "knowledge_base_name"

- 2) re-compile the shell in Turbo Prolog.

Figure 6.4 shows the basic control flow of the inference engine, which performs the function of a rule interpreter. Rules are read from the knowledge base. The inference

Figure 6.4 The Basic Control Flow of the J.shell Inference Engine

engine takes the term "initialize" as its goal, and the elements in the body of this term as the first-level sub-goals, and tries to prove these subgoals. This procedure is repeated until the sub-goal reached is either related to a fact stored in the knowledge base, or to a question whose answer can be obtained from the user. This is often referred as a recognize-act control. Meanwhile, the inference engine traces and records the reasoning path in order to be able to answer the user questions like "WHY" and "HOW". If a path fails, the inference engine will try alternative paths sequentially, until it either reaches a solution, which means that the solution path has been found, or all the paths have failed, which means that the goal cannot be proven. More information about the shell will be given in the following sections.

6.5 TMDAS Knowledge Base

When compared to the sizes of the expert systems given in Schutzer' s table of "current application domains" (1987: 139), TMDAS, with about 500 entries, can be considered as a medium size knowledge base. It is divided into 6 sub-programs to cope with memory limitation. In addition to this knowledge base, there is another graphic module developed in C offering an on-line map library as part of the user interface.

In previous chapters, the concept design and knowledge acquisition for TMDAS was discussed in detail. In this section, some technical issues related to the realization of this concept, and the representation of the knowledge are discussed. Some examples of pseudo codes are given.

Information query

The information query module is designed to acquire the information needed for a specific map design. From the technical perspective, the purpose of this module is to establish a dynamic database which contains the specific domain related knowledge that will be used, together with the cartographic knowledge needed to carry out the map design task. This dynamic database will be built on a series of interactions with the user. The validation of

all the responses from the user will be accomplished before they are saved. This validation will be done using the known information as much as possible.

Figure 6.5 shows the reasoning in the information query module, which consists of four parts: acquiring the mapping purposes, selecting the information query mode (depending on the data used), acquiring the data properties, and grouping the variables into map layers.

Acquiring the mapping purposes: this submodule is designed to acquire the information on the use of the map, which influences the map design. Because little knowledge has been codified in this aspect, there are only two factors related to it that have been implemented in the prototype: the medium of the map, that is, screen or paper; and the complexity of the map, which can be considered as a simplified representation of different types of maps. For example, a map for a school should be simpler than a map for research purposes. If more knowledge is codified later, this module can easily be extended. The results from this query will be used as control factors in the map design process.

Selecting the information query mode: there are three information query modes designed to meet the mapping requirements of different types of users. One is for the NAIS user. If this mode is selected, the system will automatically open the stand alone NAIS knowledge base which presently includes 23 NAIS variables. A list of all the variables will appear on the screen. The user can select the variable(s) needed, and the knowledge related to these variable(s) will be loaded into the dynamic database. For example, the following knowledge has been stored in the NAIS knowledge base for the variable "Land_cover":

```
fact(11, variable(6,"Land_cover")).  
fact(12, measurement_scale("Land_cover","nominal")).  
fact(13, spatial_dimension("Land_cover","total_area")).  
fact(14, statistical_map("Land_cover","n")).
```

Figure 6.5 Reasoning in the Information Query Module

These facts indicate that the spatial dimension of the variable "Land_cover" is "total_area" (covering the entire mapping area), the measurement scale is "nominal", and "Land-cover" is not a variable suitable for statistical mapping. There is no need, therefore, for additional knowledge from the user. The second mode is designed for the mapping of any other data set brought by the user, because providing cartographic advice on map design for different application domains is the main objective of this project. If this mode is selected, the user is expected to answer some questions in order to help the system establish the dynamic database which consists of the specific knowledge required to carry out the map design. This aspect will be discussed later in more detail.

The third mode is for the user who wants to use data from NAIS with data from another data set. If this mode is selected, the user can first select data from the NAIS list, then, add the data from the other data set. The user is expected to have some knowledge of the second data set.

Acquiring the data properties: when user data is used, the data properties must be acquired. In this submodule, the user is asked questions, such as the variable name, its spatial dimension and the measurement scale. The answer can be selected from menus. For example, the question for measurement scale of variable "rainfall" will be:

What is the measurement scale of the variable: rainfall

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice:

The user can select one of the choices by keying in the number of that choice. As emphasized before, in this case, it is assumed that the user has knowledge of the data and of the terminology used by TMDAS. From the system development perspective, a validation verification mechanism is also needed and has been constructed. It has two functions: 1) to check typing errors. In the previous example, any characters other than 1 to 4 will not be accepted; 2) to determine logically inconsistent errors. For example, if the user indicates that the spatial dimension of "Rainfall" is "volume", and the measurement scale of "Rainfall" is "nominal", the system will determine, after the self-verification, that there is a logical inconsistency between these two pieces of information, because, by definition, "volume" data can not have a "nominal" measurement scale (see the definition in chapter 2). Therefore, the system will conclude that at least one of the answers is wrong. Then, all information related to "Rainfall" will be removed, and all questions will be re-asked after a short explanation. The pseudo code of the rule for this problem can be written as:

```
rule(133,check_sdms(Is,Variable,Sd,Ms),
    [myeqc(Sd,"volume"),
    myeqc(Ms,"nominal"),
    say_sound,
    mywrite("Note: to portray volume data, quantitative"),
    mywrite("data has to be used."),
    myretract(spatial_dimension(Variable,_)),
    myretract(measurement_scale(Variable,_)),
    no_variables(No),
    get_variable(Is,_,No)]).
```

This verification facility is helpful, but it does not guarantee that all the information is correct. That is why the assumption of the user needs must be emphasized. More verification facilities of the system will be discussed in the next submodule.

Grouping variables into map layers: map layer has been introduced as the basic unit for symbol design. The principles applied to define a layer and the benefits of using map layers have already been discussed. The following discussion will focus on the issue of applying these principles.

Defining the relationships between variables in a data set means defining the relationships between ANY TWO variables in the data. For example, if a seven-variable data set is being dealt with, the number of relationships will be:

$$C_7^2 = 7! / 2! * (7-2)! = 21$$

But it would be very disturbing to ask a question 21 times. Actually, the problem can be simplified through the use of the semantic net introduced in chapter 3

C: Consist_of
N: Neighbor_of
R: Related_to

Figure 6.6 Example of the Data Relationships Represented by a Semantic Network

Figure 6.6 gives an example of the relationships represented by a simple semantic net. As already discussed, a key feature of the semantic net is that it can establish a property

inheritance hierarchy in the net. In the given example, if the facts that "variable A consists of variable B, and variable A is related to variable F" are known, then the fact that "variable B is related to variable F" can be inferred according to inheritance property. Based on this principle, if the relationships represented in the figure by solid lines are known, all other relationships represented by dashed lines can be derived. A semantic net can be represented in Turbo Prolog by using facts and rules. Normally, facts are used to represent the semantic network, and rules are used to describe the inheritance property. In the TMDAS knowledge base, the previous semantic net can be represented using the defined format. Some facts:

```
fact(220,consist_of (a,b)).  
fact(221,consist_of (a,c)).  
fact(224,related_to (a,f)).  
fact(225,related_to (a,g)).
```

are used to describe the net, and rules:

```
rule(221, related_to(X,Y),  
    [consist_of (A,X),  
     related_to(A,Y)]).  
  
rule(222, neighbour_of(X,Y),  
    [consist_of (A,X),  
     consist_of (A,Y)])
```

are used to describe the inheritance property. In this way, the system can limit the number of questions to a minimum and establish an entire data relationship network.

The next task of the submodule is to verify the user selected relationships against the related knowledge and information, in order to determine the validity of the answer. Because both the concept of the relationship between variables and the terminology used in this system may be unfamiliar to a new user, this function is especially important. Since each relationship is strictly defined, it is easier to establish a series of verification criteria. For

example, the rules used for verifying "neighbour_of" relationship are written in the following manner:

```
rule(209, process_layer(neighbour_of, Variable1, Im, Variable2, It, No),  
    [check_sta(Variable1, Variable2),  
     measurement_scale(Variable1, Ms),  
     measurement_scale(Variable2, Ms),  
     myeq_c(Ms, "absolute_q"),  
     myretract(map_layer(Im, Ni, Itl)),  
     combine(Variable2, Itl, New_list),  
     increase(Nn, Ni),  
     myassertz(map_layer(Im, Nn, New_list)),  
     increase(Itl, It),  
     group_layers(Im, Itl, No)]).
```

```
rule(220, check_sta(Variable1, Variable2),  
    [stati_map(Variable1, n),  
     stati_map(Variable2, n),  
     spatial_dimension(Variable1, point),  
     spatial_dimension(Variable2, point)]).
```

```
rule(221, check_sta(Variable1, Variable2),  
    [stati_map(Variable1, y),  
     stati_map(Variable2, y)]).
```

The basic principle applied here is that if variable A and variable B have the "neighbour_of" relationship, they have the same data properties and the data must be at the absolute quantitative measurement scale. The above rules can be explained through the following checks and actions: 1) whether both variables are used for statistical mapping; if not, whether they have the same spatial dimension (check_sta(Variable1, Variable2)); 2) whether they have the same measurement scale; 3) whether the measurement scale is absolute quantitative. If the above questions are all proven true, the system will perform the following actions: 1) remove the layer created for variable B; 2) add variable B to the layer which contains variable A; 3) continue with the next variable.

Similar rules have been developed for all relationships, according to their definitions. Therefore, if an incorrect relationship is selected, the user may get a warning sound followed by a message such as:

Based on the available information
TMDAS found that the relationship between:
Variable_a, and, Variable_b can not be: R

where Variable_a and Variable_b will be the actual names of the variables, and the R will be the relationship selected. The user will be asked to re-select the relationship.

The last task that this submodule performs is to verify the map capacity as a first level control; this must be re-done after map types have been selected. Two kinds of controls are applied here. One is to verify if there are graphic representations which can support the defined relationships. For example, a compound point symbol (e.g. pie-chart) and a "meaningful combination" (refer to chapter 2) point symbol are allowed in the system, but, in general, it may not be recommended to apply both to the same point symbol, because the same visual variable must be used in both cases. In this case, the overall map capacity may not be a problem, but this combination may exceed the ability of the graphic representation of this specific component. For this kind of situation, the following message will appear:

Sorry! Based on the previously entered data
TMDAS found that it can not combine:
Variable_a, and, Variable_b,
into one map for the defined relationship: R
Re-arrange the data, and start again!
Press any key to continue.

The second control applies when the number of map layers exceeds the maximum layer defined by the purpose of the map. If this is the case, the following message will appear:

Sorry! Based on the previously entered data
TMDAS found that it can not combine
all data into one map
Re-arrange the data, and start again!
Press any key to continue.

In this section, some important issues implemented in the four submodules of the information query module have been introduced. If this module is successfully executed, a dynamic data base which contains all the specific knowledge for this mapping task will be established and used for the reasoning in the modules described below.

Map type selection

Map type selection is considered an important intermediate procedure in TMDAS. There are three tasks defined in the concept design for this module:

- 1) assist in finding the proper symbol type for given variables;
- 2) assist in checking the possibility of combining multiple layers on one map;
- 3) assist in identifying alternative map types based on the relationships between map types.

The taxonomies of map types defined for this purpose have been discussed in chapter 5. The implementation of these taxonomies and their uses in fulfilling the three functions will be discussed in this and the next section.

As shown in figure 6.7, the task of this module is to select the proper map types for each of the map layers created in the first module, and to relate the map types with symbol types.

Since the hierarchical map type categories used are based on both the data properties, which have already been stored in the dynamic data base, and the symbol types, the rules

Figure 6.7 Reasoning in the Map Type Selection

can be developed directly. For example, for a point map type and a line map type, the rule can be:

```
rule(336,layer_type(L,point),
    [layer_sd(L,point)]).
```

```
rule(337,layer_type(L,line),
    [layer_sd(L,line)]).
```

For statistical mapping, either a choropleth map or an area-based-point map (refer to chapter 5) can be selected, according to the measurement scale of the data. The rules will be:

```
rule(346,layer_type(L,choropleth),
    [map_layer(L,1,[Variable]),
     stati_map(Variable,y),
     layer_ms(L,derived_q)]).

rule(348,layer_type(L,area_based_point),
    [map_layer(L,1,[Variable]),
     stati_map(Variable,y),
     layer_ms(L,absolute_q)]).
```

Because each map type is related to only one symbol type, facts can be used to depict the relationships:

```
fact(402,symbol_type(point,point)).
fact(401,symbol_type(line,line)).
fact(404,symbol_type(area_based_point,point)).
fact(411,symbol_type(area,area)).
fact(413,symbol_type(choropleth,area)).
```

The selection of map type and symbol type are carried out layer by layer. As a result, the most suitable map types for each layer are selected based on the acquired data properties, regardless of the effect on other layers. This problem is solved in the next module.

Map type combination

The map type combination module is designed to verify if it is permitted to combine all the layers into one map sheet using the selected map types. There are two types of checks needed. Firstly, whether any two layers can be used together without creating a misleading or disturbing presentation; secondly, whether the entire map content exceeds the capacity

Figure 6.8 Reasoning in the Map Type Combination

of the map – this is the second control level for map capacity. The reasoning procedure is shown in figure 6.8. In the first type of check, each layer has to be verified against all the others. The possible result can be, as defined in chapter 5, permitted (p), permitted with restriction (pwr), or not recommended (nr). The knowledge can be represented using facts, as follows:

```
fact(301,layer_combine(point,point,p)).  
fact(302,layer_combine(point,line,p)).  
fact(318,layer_combine(area,area,pwr)).  
fact(320,layer_combine(area,category,pwr)).  
fact(334,layer_combine(category,choropleth,nr)).  
fact(332,layer_combine(category,isoline,nr)).
```

If the result is "p", the check moves to the next layer. If it is "pwr", the system has to identify the restriction. Here, the following representation is used:

```
fact(353,conflict(area,area,hue)).
```

This means that if an area symbol layer is combined with another area symbol layer, the visual variable hue can only be used once. This restriction must be recorded and will be used later for visual variable selection. If the result of the combination check is "nr", the result can be one of the two cases. First the two types of graphic representations can affect one another. The result can be misleading or disturbing. Therefore, an alternative for either map type should be sought using the following statement:

```
fact(345,layer_change(choropleth,category,dot,absolute_q)).
```

This fact indicates that "when a choropleth map layer is encountered with a category map layer, the data used for the choropleth map should be re-processed to absolute quantitative data, and then represented by the dot map method". Two important points should be emphasized: 1) when the map type of any layer is changed, the combination of all other map layers against this layer must be re-checked; 2) if the data need to be processed to

change the measurement scale, this change must be recorded in order to inform the user when the recommendations are given.

The second case for an "nr" result is due to the fact that the two kinds of graphic representations used represent two kinds of data which have inherently different characteristics, such as a category map and an isoline map, and are seldom combined into one map. If this is the case, the user may need to reorganize the data for the map design. The system will, then, automatically separate one of the two layers from the layer set, and continue the reasoning on the design with these N-1 layers. The final recommendation will include advice on this layer separation.

The second task of this module is to verify map capacity. The first level of the map capacity check was done by checking the number of layers. But, as discussed before, the capacity of the map is related not only to the number of layers, but also to the symbol types applied to the layers. For example, a paper map with three layers is generally possible, but the map capacity will be exceeded if all three layers are choropleth map layers. If this happens, the following message will appear on the screen:

*****~*****

Sorry! Based on the previously entered data
TMDAS found that it can not combine
all data into one map
Re-arrange the data, and start again!
Press any key to continue.

Visual dimension selection

The selection of the visual dimension deals with two parts, as shown in figure 6.9: the visual variable and the visual constant.

The use of the visual variable relies on two factors, the measurement scale of the data and the symbol type used. As discussed in chapter 5, there are some practical restrictions for

Figure 6.9 Reasoning in the Visual Dimension Selection

using certain visual variables with certain symbol types. For example, the rules for applying orientation and spacing would be:

```
rule(410,visual_variable(L,orientation),  
    [layer_rms(L,nominal),  
     layer_symbol_type(L,point)]).
```

```
rule(413,visual_variable(L,spacing),  
    [layer_rms(L,ordinal),  
     layer_symbol_type(L,area)]).
```

Rule 410 indicates that the visual variable orientation can be applied to the layer whose measurement scale is nominal and whose symbol type is point. In other words, orientation would not be recommended if the layer uses either line or area symbols. Similarly, spacing is only recommended to represent ordinal data with area symbols.

As discussed previously, for some types of perception properties, there can be more than one visual variable available. Depending on the circumstances, one visual variable may be preferable. For example, both value and spacing can be applied to a choropleth map, but value is more often used for screen-based maps, and spacing for plotted maps (because of the convenience in producing maps). But, this is only relative. For this reason, the system is designed to give the user all the correct choices listed in a best-first order. Therefore, the user can, according to one's experience, either use the first one, which is the best option suggested by the system, or make another choice for a specific reason. In either case, the correctness of the map is guaranteed. To realize this procedure in J.shell, an enforced backtracking is needed, which means that after a solution is found and recorded, it is necessary to "deceive" the system by telling it that the solution is "false". The system will then try to find the second solution. This can be repeated until all solutions are found. In Turbo Prolog, the built-in predicate "fail" can be used for this purpose. A pseudo code of this is shown below:

```

rule(427,layer_visual_dimension(L,Vv,Vc,No),
    [less_e(L,No),
     visual_variable(L,Vv),
     check_restriction(L,Vv),
     visual_constant(Vv,Vc),
     increase(Ivn,Iv),
     myasserta(layer_vv(L,Ivn)),
     myassertz(layer_visual_feature(L,Ivn,Vv,Vc)),
     myfail]).

```

In this rule, "myfail" (which accesses the built-in predicate "fail") will fail the rule at the end, so that backtracking is enforced. The statement in the rule, "check_restriction(L,Vv)", is used to call another rule to check if there is a restriction for layer L using Vv as visual variable.

For each visual variable recommended, a visual constant also needs to be recommended. The statement "visual_constant(Vv,Vc)" in the previous rule accesses facts such as:

```
fact(416,visual_constant(hue,[value,size])).
```

If Vv is matched with *hue*, then Vc will be matched with *value* and *size*, which means that if hue is used as a visual variable, value and size should be kept constant.

6.6 Explanation Facility in The Inference Engine

The explanation facility is an important characteristic of expert systems. Its basic concepts and functions have previously been discussed. In this section, details on these functions in J.shell are introduced through an example of a simple knowledge base, constructed for demonstration purposes:

```

fact(1, spatial_dimension(ottawa,point)).

rule(1,symbol_type(X,point),
    [spatial_dimension(X,point)]).

```



```

rule(2,visual_variable(X,size),
    [spatial_dimension(X,point),
     measurement_scale(X,absolute_quantitative)]).

rule(3,recommendation(X,Symboltype,Visualvariable),
    [symbol_type(X,Symboltype),
     visual_variable(X,Visualvariable)]).

rule(4, initialize,
    [recommendation(ottawa,Symbol_type,Visual_variable)]).

```

This is a simple knowledge base constructed according to the J.shell format. It has one fact and four rules. Because the term "initialize" has been reserved for the goal, the reasoning will start from rule 4 by unifying X with "ottawa". The recognize-act control in J.shell makes it access rule 3, then rule 1 and 2 in turn. Because the body of rule 1 is a fact, then this path has been found to be true. But whether "measurement_scale(ottawa, nominal)" is true can not be proven from the knowledge base, so the user will get a question like:

Does the variable: ottawa

has the measurement scale of: absolute_quantitative (y/n)

If the user wonders why this question is asked, the WHY facility can be used by typing "w". The following message will appear on the screen:

J.SHELL is trying to determine whether the following
rule is applicable in this consultation :

Rule 2:

IF spatial_dimension(ottawa,point),
 measurement_scale(ottawa,absolute_quantitative),
THEN visual_variable(ottawa,size)

The function of this WHY facility is to indicate to the user the context in which this piece of information is needed. For ease of understanding, the rule is represented in the "IF...THEN..." format.

If the answer to the question is positive, the reasoning will be completed, followed by a recommendation such as:

variable	symbol_type	visual_variable
ottawa	point	size

Further, the user may want to know how the result has been reached. Here, the HOW facility can be used. Because J.SHELL has traced the reasoning process, it can present the solution path that the inference engine followed. There are two ways to present this reasoning. One is called *static*, and it shows all the states in the backward-chaining order at once, as follows:

```

*****
[initialize, recommendation("ottawa","point","size"),
visual_variable("ottawa","size"),
measurement_scale("ottawa","absolute"),
spatial_dimension("ottawa","point"),
symbol_type("ottawa","point"),
spatial_dimension("ottawa","point")]
*****

```

The other way to demonstrate the reasoning is called *dynamic*, and it conceptualizes the reasoning as a tree, and presents the states one by one from leaf to root as shown in figure 6.10.

Figure 6.10 The Inference Tree Based on the Example Knowledge Base

By giving this true inference path rather than a simple pre-organized text, the user is able to judge, whether the recommendation can be accepted, and, if not, whether there are mistakes in either rules or facts.

6.7 On-line Map Library

If the reasoning is accomplished successfully, the user may get a final recommendation similar to the one shown in figure 6.3, which indicates symbol types, visual variables, and visual constants to be used for the map. But, since the user may be cartographically untrained, terms like hue, value, etc. may not be familiar. Therefore, an on-line map library is designed to explain these terms through example maps. For each of recommendation, there are some corresponding example maps available through the *view* function. Some of these maps are found in the appendix. There are two important aspects in the use of this on-line map library. 1) These maps are just example maps. TMDAS does NOT yet include the function to access the user's data. 2) For a multi-layer map, the *view* function only allows example maps to be viewed layer by layer. Hence, the user should be aware that the final map will be an overlay of all the single layers.

6.8 Summary

This chapter demonstrated the use of the technology introduced in chapter 3 for representing the knowledge acquired in chapter 2 and chapter 5, in order to realize the concept design discussed in chapter 4. Some technical issues of Turbo Prolog were also introduced, because Turbo Prolog is the programming language used in the development of J.shell. The basic formats, the inference engine, and the explanation facilities in J.shell have been discussed in detail. The TMDAS knowledge base has been presented module by module. Some key technical issues and knowledge representation in each module have been addressed with examples and some pseudo codes have been presented to give an overview on the internal structure of the system.

7. CASE TESTS AND ANALYSIS

7.1 Introduction

It has been emphasized that exploratory programming is one of the distinctive characteristics in expert system development. As stated, "expert system design and development must be carefully programmed if success is desired" (Castillo and Alvares, 1991: 19). There are ten steps in the development of an expert system:

- "(1) statement of the problem to be solved**
- (2) searching for a human expert or the equivalent data or experience**
- (3) design of the expert system**
- (4) selection of the degree of participation of the user**
- (5) selection of the development tool, shell or programming language for development**
- (6) development of a prototype**
- (7) prototype checking**
- (8) refinement and generalization**
- (9) maintenance**
- (10) updating" (Castillo and Alvares, 1991: 19).**

Steps 1 to 6 in the development of TMDAS have been discussed in previous chapters. This chapter introduces the work done in regard to step 7 and 8.

System testing can be separated into different levels. A variety of tests were carried out during system implementation. The tests to be discussed in this chapter are selected from those performed after system refinement. The main purpose of these tests is to validate the knowledge base by using real cases. In addition, these tests are also used to check system efficiency (e.g., the execution speed of the system), system sufficiency (e.g., the range of thematic mapping requirements supported), and the overall functionalities of the system.

The cases introduced below are selected from the National Atlas of Canada; seven of them are from the Fifth Edition, which was produced by the Geographical Services Division.

Canada Centre for Mapping, Energy, Mines and Resources Canada, and one (case 7) from the Fourth Edition, which was published by the Macmillan Company of Canada Limited, Toronto, Ontario, and produced by the Geographical Branch of the Department of Energy, Mines and Resources, and Information Canada. They can be divided into two types and involve the use of the three information query modes in TMDAS: variables that have been stored in the NAIS knowledge base such as the ones in case 1 and 4, variables that have not been stored such as those in case 2, 3, 5, 7, 8, or the combination of the both such as case 6. The variables which have not been stored can be considered as a user-brought data set. All maps are assumed to be designed for scientific analysis purposes and printed on paper. Different types of maps have been selected for checking the system in as many aspects as possible. There were about 25 maps tested and 8 unique cases are described below.

Each case will be demonstrated in three parts. The first part introduces the data set with a brief description of the properties of the data to be mapped and the purpose of the map. Based on this description, a cartographer may be able to make a map design. The source of the map is given. The second part presents the TMDAS recommendation. This is followed by the last part, the result analysis. The actual running procedures for all cases are captured, and are given in the appendix.

7.2 Cases and analysis

Case 1: CANADA LAND COVER ASSOCIATIONS

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1989.

Introduction

The variable represented in this map is *land cover association* based on a satellite imagery interpretation. There are twelve major cover classes identified: forest, taiga, tundra, sparse vegetation, barren land, wetland, water, ice, tidal marine areas, rangeland, and built up areas. The description of the data properties using TMDAS terminologies would be:

variable(1, land_cover)

spatial_dimension(land_cover, total_area)

measurement_scale(land_cover, nominal)

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Land_cover 1	area/ category	hue form	value size, value, spacing

Analysis

TMDAS has recognized that this will be a single layer map with one variable, and has recommended that the area symbol should be applied with either the hue or form visual variable.

In the actual map, hue and area symbols are applied, because hue has better selective perceptions than form, especially when the map has many classes. The map type in the recommendation indicates the name of the map type used in TMDAS; this does not affect the map design.

In the actual map, the "secondary cover class" is represented through the use of letters. The mapping of hierarchical nominal data is not yet supported by TMDAS.

Case 2: CANADA FARM OPERATORS

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1985.

Introduction

There are three variables represented in this map. The first is *farm operator*, defined as the person who is directly responsible for the agricultural operation. The second is *percentage change in the number of farm operators between 1971 and 1976 by census division*. And the third, termed *agricultural activity*, is used to indicate the areas where such activity occurs, but the agricultural area is too small to be shown. This data set can be described in TMDAS terminologies as follows:

```
variable(1, farm_operators)
```

```
spatial_dimension(farm_operators, discrete_area)
```

```
measurement_scale(farm_operators, absolute_quantitative)
```

```
variable(2, farm_land_change)
```

```
statistical_map(farm_land_change, yes)
```

```
measurement_scale(farm_land_change, derived_quantitative)
```

```
variable(3, agricultural_activity)
```

```
spatial_dimension(agricultural_activity, point)
```

```
measurement_scale(agricultural_activity, nominal)
```


TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Farm_operators 1	dot/ dot	*	form, hue
2	Farm_land_change 1	area/ choropleth	spacing value value_hue	form, hue, size hue
3	Agricultural_activity 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue

Analysis

TMDAS has recognized that this should be a three layer map, with each of the layers having one variable. For the farm operators layer, the dot method is recommended for the distribution of this discrete feature. The choropleth method is recommended for the changes to farm land, through an area symbol with one of three visual variables, in which, value-hue is a redundant combination of value and hue (introduced in chapter 2). For the last layer, agricultural activity, a point symbol is suggested with the selection from three visual variables.

This recommendation concurs with the graphic representation used in the actual map. In the map, the variable farm operators is portrayed by dots, and two types of dots are used with different values assigned. A choropleth map layer is used for the changes in farm land. An area symbol with the redundant combination of the visual variables value and hue, value-hue, is used to describe the changes in two directions (as discussed in chapter

2), decrease and increase. A point symbol is used for agricultural activity to indicate the presence of farm lands.

Case 3: CANADA SAWMILLS

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1992.

Introduction

Three variables are represented in the map Canada Sawmills. The first is *lumber production* at sawmills, using "thousands of cubic meters" as the measurement unit. This is the estimated total production of lumber for 1983. The second variable is *wood type*, which indicates principal tree types processed at a particular mill. The third is *forest area* which represents the forest areas of commercial significance. The forest areas have been divided into two types: forest, and forest with agricultural and cleared lands. This data set can be described in the TMDAS format as follows:

```
variable(1, lumber_production)
```

```
spatial_dimension(lumber_production, point)
```

```
measurement_scale(lumber_production, absolute_quantitative)
```

```
variable(2, wood_type)
```

```
spatial_dimension(wood_type, point)
```

```
measurement_scale(wood_type, nominal)
```

```
variable(3, forest_area)
```

```
spatial_dimension(forest_area, partial_area)
```

```
measurement_scale(forest_area, nominal)
```

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Lumber_production 1	point/ point	size	form, hue
1	Wood_type 1	point/ point	hue	value, size
2	Forest_area 1	area/ area	hue form	value size, value, spacing

Analysis

TMDAS has recommended the exact graphic representation methods used in the actual map. Although there are three variables in the data set, TMDAS has determined that there will only be two map layers involved, with one of the layers containing two variables, lumber production, and wood type. Because these two variables are applied to the same locations, and use different visual variables, a "meaningful combination" (introduced in chapter 2) of the two visual variables can be employed in one point symbol to represent them. This point symbol is found on one map layer. An area symbol, with either hue or form, is recommended for the representation of forest areas.

In the actual map, size and hue are combined and applied to the point symbols for representing the first two variables, except that the size is shown in class intervals, probably in consideration for the legibility of the map. Forest area is depicted by area symbols with hue as the visual variable.

Case 4: CANADA OIL PIPELINES

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1984.

Introduction

The information presented in this map is the location of major pipelines for the transmission and distribution of oil in Canada. There are three types of data included: *oil pipelines*, *pump stations* on major lines, and associated *oil fields*. In TMDAS formats, they can be represented as follows:

`variable(1, oil_pipelines)`

`spatial_dimension(oil_pipelines, line)`

`measurement_scale(oil_pipelines, nominal)`

`variable(2, pump_stations)`

`spatial_dimension(pump_stations, point)`

`measurement_scale(pump_stations, nominal)`

`variable(3, oil_fields)`

`spatial_dimension(oil_fields, p_area)`

`measurement_scale(oil_fields, nominal)`

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Oil_pipelines 1	line/ line	hue	value, size
2	Pump_stations 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue
3	Oil_fields 1	area/ area	hue form	value size, value, spacing

Analysis

TMDAS has suggested that this map should have three map layers, each of which has one variable. For these three layers, line, point, and area symbols have been recommended respectively. Because they are all nominal data, the visual variables related to the representation of nominal data are recommended. However, as was discussed before, the possible selections for each of these three types of symbols are different .

In the real map, the visual variable hue is not employed for oil pipelines. Instead, two types of line symbols, solid line and dashed line, are used to represent existing pipelines and proposed pipelines respectively, according to the cartographic conventions. The visual variable hue is applied to area symbols, and the point symbol to pump stations.

Case 5: CANADA POPULATION DISTRIBUTION 1976

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1985.

Introduction

The information presented in this map is the population distribution in Canada in 1976, based on the 1976, Census of Canada. There are three types of data used: *population at populated places*, *population by enumeration areas*, and *isolated settlements*. In TMDAS formats, they can be represented as follows:

`variable(1, place_population)`

`spatial_dimension(place_population, point)`

`measurement_scale(place_population, absolute_quantitative)`

`variable(2, discrete_population)`

`statistical_map(discrete_population, yes)`

`measurement_scale(discrete_population, absolute_quantitative)`

`variable(3, isolated_settlement)`

`spatial_dimension(isolated_settlement, point)`

`measurement_scale(isolated_settlement, nominal)`

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Place_population 1	point/ point	size	form, hue
2	Discrete_population 1	dot/ dot	*	form, hue
3	Isolated_settlement 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue

Analysis

TMDAS has recognized that this map will require three map layers with differing symbol designs. One uses proportional point symbols for population at populated places. Another employs dots to portray discrete population. And the third uses point symbols to indicate isolated settlements with one of three visual variables.

A common mistake of novice map designers in mapping variables like discrete populations is often the use of a choropleth map. It can frequently be observed that absolute quantitative data by enumeration areas is represented by a choropleth map, in which the combination of the varying sizes of the statistical units and their mapped values alters the impression of the distribution of the phenomena.

In the actual map, a proportional point symbol is used for population at populated places. Hue is also applied to enhance the legibility of the point symbol, and, as is recommended by TMDAS, it is applied as a visual constant. Discrete population is represented by the dot

method, with 5 types of dots with different values. Point symbols are used in the representation of isolated settlements.

Case 6: CANADA SOLAR RADIATION-ANNUAL

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1984.

Introduction

Three variables are represented in this map. The first is *climate stations*. The second is *annual mean daily global solar radiation*, which is the average of the daily global solar radiation values for one year. The third is *annual mean daily global solar radiation variability*, which is the standard deviation of the annual mean daily global solar radiation.

This data set can be described in the TMDAS format as follows:

```
variable(1, climate_station)
```

```
spatial_dimension(climate_station, point)
```

```
measurement_scale(climate_station, nominal)
```

```
variable(2, solar_radiation)
```

```
spatial_dimension(solar_radiation, volume)
```

```
measurement_scale(solar_radiation, derived_quantitative)
```

```
variable(3, radiation_variability)
```

```
spatial_dimension(radiation_variability, volume)
```

```
measurement_scale(radiation_variability, derived_quantitative)
```


TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	climate_station 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue
2	Solar_radiation 1	isoline/ isoline	*	form, hue
3	radiation_variability 1	isoline/ isoline	*	form, hue

Analysis

Global solar radiation and its variability are all continuous volume data, which is another type of data that should not be mapped with a choropleth map, since its distribution is not controlled by political or administrative subdivisions. TMDAS has recommended use of the isoline method for these two variables and nominal point symbols for climate stations.

These are also the graphic representations used in the actual map, except that layer-tint (refer to chapter 5) has been used together with the isolines for global solar radiation, in order to enhance visual perception.

Case 7: CANADA MONTHLY DISTRIBUTION OF RUNOFF

This map is selected from the National Atlas of Canada, Fourth Edition. It was printed in 1974.

Introduction

There are 12 variables in this map, which are the *percentages of volume of flow by months as percentages of the annual average volume of flow at selected gauging stations of representative rivers*. This data set can be described in the TMDAS format as follows:

variable(1, jan_runoff)

spatial_dimension(jan_runoff, volume)

measurement_scale(jan_runoff, derived_quantitative)

periodically_changed(jan_runoff, yes)

variable(2, feb_runoff)

spatial_dimension(feb_runoff, volume)

measurement_scale(feb_runoff, derived_quantitative)

periodically_changed(feb_runoff, yes)

The same representation can be repeated for the remaining ten months.

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
	Jan_runoff Feb_runoff Mar_runoff Apr_runoff May_runoff Jun_runoff Jul_runoff Aug_runoff Sep_runoff Oct_runoff Nov_runoff Dec_runoff			
1	12	special_graphics/ located_graphics	size	form, hue

Analysis

Although there are 12 variables in this data set, TMDAS has detected that there should be only one map layer involved, which includes all the variables and uses compound point symbols. Runoff is a type of volume data which changes periodically. These point symbols at selected gauging stations are used to portray the changes in the phenomena over regions. TMDAS refers to this type of map as a located graphics map.

These are also the graphics used in the actual map. The special point symbol applied is a bar-chart. The visual variable size, in this case it is the height, is used.

Case 8: CANADA THE AGING POPULATION

This map is selected from the National Atlas of Canada, Fifth Edition. It was printed in 1993.

Introduction

This map depicts the age structure of the Canadian population in 1986 with emphasis on the elderly. The data used is the *total number of population age 65 or over*, and it is divided into two groups: *65 - 74, 75 and over*, for each census division. The *boundaries of census divisions* are shown. This data set can be described in the TMDAS format as follows:

```
variable(1, total_elder)
```

```
statistical_map(total_elder, yes)
```

```
measurement_scale(total_elder, absolute_quantitative)
```

```
variable(2, 65_group)
```

```
statistical_map(65_group, yes)
```

```
measurement_scale(65_group, absolute_quantitative)
```

```
variable(3, 75_group)
```

```
statistical_map(75_group, yes)
```

```
measurement_scale(75_group, absolute_quantitative)
```

```
variable(4, census_boundary)
```

```
spatial_dimension(census_boundary, line)
```

```
measurement_scale(census_boundary, nominal)
```

TMDAS recommendation

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	75_group 65_group Total_elder 3	compound_point/ area_based_point	size	form, hue
2	Census_boundary 1	line/ line	hue	value, size

Analysis

TMDAS has grouped the four variables into two layers. The first three form one layer, in which the compound point symbol with the visual variable size are recommended to make an area-based point map; because absolute quantitative data is involved, a choropleth map should not be used. For the census division boundaries, a nominal line symbol is suggested.

This recommendation represents the same graphics used in the actual map. But, in addition, there is another variable, the percentage of the elder population in the total population, represented in each census division. The percentage is classified into four groups: < 4.0, 4.0-6.9, 7.0-14.0, >14.0. A combination of the visual variables value and hue, value-hue, is used to represent this data, and is tinted within the compound point symbols which are used for the total elder population of regions. The integrated symbol forms a "meaningful combination": the size of the symbol refers to the total number of elderly in the population, the sectors refer to the proportion of elderly in sub-groups, and the value-hue applied to this symbol refers to the percentage of the elderly population in the

total population of the region. This is a very efficient graphic representation because it reveals several types of information in one integrated symbol. Yet, as can be seen, the data relationships between these variables are complex. This type of integrated symbol must be used carefully to avoid misleading results. Hence, it is not supported by TMDAS. In this case, TMDAS would give the information indicating that the relationships between variables are unacceptable.

7.3 Summary

In this chapter, 8 real cases, selected from preliminary testing using the National Atlas of Canada, Four Edition and Fifth Edition, have been introduced. There were other tests carried out using data from these atlases, such as other variables in the NAIS data set list (refer to the appendix) and other maps which includes: Canada – Heating Degree-Days, Canada – Last Frost in Spring, Canada – Temperature – April and October, Canada – Distribution of Wetlands, Canada – Wetland Regions, Canada – The 32nd Parliament, Canada – Fisheries Resources, Canada – Natural Gas Pipelines, Canada – Electricity, 1987, Canada – Telecommunications Systems, and Canada – The Potential of Soils and Bedrock to Reduce the Acidity of Atmospheric Deposition. The cases selected here show different map types. Each of the cases is different from the others in at least one of the following aspects: the relationships between variables, the map types involved, the number of map layers, and the map layer combinations. The cases, therefore, can be considered as representatives of different types of map designs. The cases also involved the use of different information query modes. The preliminary test has given positive feedback on the feasibility of TMDAS for both the concept design and the knowledge base. TMDAS supports most mapping requirements with acceptable results. On the other hand, it can also be seen that, as predicted, the system can not work as flexibly as human cartographers, and it has limitations. The system can not support the representation of hierarchical nominal data structures, and it may not permit a complex symbol combination, even though this

may be a very good result as in case 8, due to the use of conservative restrictions. It can be concluded that the system is a useful tool for most thematic map design tasks, because most maps do not exceed the complexity supported by TMDAS.

TMDAS, however, is still in the early stages of development. The case tests introduced in this chapter are preliminary tests carried out by the author, the system developer. Many more tests are required by both cartographic experts and end-users in order to identify its imperfections, to improve the user interface and to enlarge the knowledge base. This stage may be quite lengthy. For example, in the development of MYCIN, the expert system in the domain of medicine, developed in 1970's, "about 50 person-years were expended in the effort" (Luger and Stubblefield, 1989: 315). To develop a fully operational expert system in thematic map design, much effort is required.

8. CONCLUSION

8.1 Summary and Conclusions

This research project has reached its initial objectives: acquiring the knowledge inherent in thematic map design, for the purpose of providing this knowledge in an automated mapping environment designed to safeguard against the construction of misleading maps through the use of expert system technology. The research was carried out as a knowledge engineering project, in which the concept design was developed for an expert system in thematic map design; the knowledge was codified, further developed, and transformed into formal knowledge; this knowledge was structured into modules and knowledge representation schemes for different types of knowledge were studied and applied; an expert system shell, named J.shell, was developed; the TMDAS knowledge base, with about 500 knowledge entries, was implemented in J.shell format; and finally, preliminary case testing and analysis was accomplished.

Concept Design: The development of a fully operational thematic map design advisory system is a long term engineering project, and an incremental approach must be applied in the development of such a system. The concept design is, therefore, important for final success. This aspect was thoroughly analyzed. Based on the analysis of user requirements and the characteristics of the theme oriented approach, applied in other thematic map design expert systems that were reviewed, a different approach – the cartographic function and data property oriented approach – was proposed. The feasibility of this approach and its application were discussed through an analysis of the manner in which human cartographers solve map design problems. The project presented two frameworks based on the proposed approach, the optimal and the modified framework. Four of the submodules, the information query module, the map type selection module, the map type

combination module, and the visual dimension selection module, were implemented in the prototype of TMDAS.

Knowledge acquisition: As for all AI projects, knowledge acquisition is one of the key issues in the development of TMDAS. The knowledge acquisition in TMDAS involved two types of research. The first was the codification and extraction of knowledge in map design from selected authoritative textbooks, to establish a knowledge frame and provide a solid background for TMDAS. The other research dealt with further study of the components of the knowledge where mature knowledge was not available, in order to form a complete body of knowledge for the development of TMDAS. The second part of the research was carried out mainly based on the analysis of the design of selected maps and atlases. In addition, knowledge from some of the NAIS data set, considered as a specific data set, was extracted to form a stand alone NAIS data knowledge base. In the knowledge acquisition procedure, 14 knowledge acquisition tables were designed, to help codify and transform the knowledge into formal knowledge.

Knowledge modules and representation: Since a large amount of knowledge is involved in the development of TMDAS, knowledge modules and knowledge representation schemes were carefully studied. The knowledge was structured according to the system concept design, and knowledge representation for different types of knowledge has been discussed.

The development of J.shell: Based on the analysis of available system resources and the configuration requirements of TMDAS, an expert system shell, J.shell, has been developed for this project. J.shell is a simple and generic shell developed on an IBM-PC compatible computer, programmed in Turbo Prolog. Any knowledge base, written according to the format defined by J.shell, can use J.shell as the inference engine. The explanation facilities of expert system such as "WHY", "HOW", and "Explain" have been implemented in J.shell. Because of the use of J.shell in the project, the TMDAS knowledge includes domain related knowledge only, hence, it is easy to update.

The implementation of the TMDAS knowledge base: The TMDAS knowledge base was implemented in the J.shell knowledge representation format. It contains about 500 entries of knowledge and has been divided into 6 submodules. One of them is a stand alone NAIS knowledge base. The rest is composed of generic knowledge in map design, which includes both theoretical principles, such as the rules of graphic semiology developed by Bertin, and heuristics, such as map type selection and map type combination control. For each mapping requirement, a dynamic data base will be established based on an interaction with the user. This data base includes the specific knowledge needed for this specific mapping requirement. Therefore, TMDAS knowledge base supports a variety of mapping requirements in a generic way.

Case tests and analysis: Preliminary case tests and analysis on the use of the prototype of TMDAS were accomplished. The cases introduced in the thesis were selected from the National Atlas of Canada and covered various types of thematic map design. The recommendations given by TMDAS were compared with the design used for the maps in the Atlas. The result proves that TMDAS supports most of the common thematic map designs dealing with multiple variables, multiple feature types and multiple symbol types, although it has limitations in handling hierarchical nominal data as well as in supporting complex variable relationships.

In comparison with previous studies in the domain of expert systems in thematic map design, the innovations in this project can be summarized in the following four aspects:

1. *The cartographic function and data property oriented approach in concept design.* Most of the previous systems used the so-called theme-oriented approach, which makes the systems domain dependent, hence, the use of the systems is very limited. In order to overcome the limitation of the theme-oriented approach, the cartographic function and data property oriented approach has been proposed, based on the analysis of user requirements. There are two main characteristics to this approach. Firstly, the modules in the approach

are divided into a limited number of cartographic functions, therefore, an incremental approach towards the final integrated system can be applied. Secondly, the reasoning is based on generic knowledge of data properties rather than specific knowledge of a defined data set; the system is therefore domain independent.

2. *The specification of a hierarchical map type category.* There was little research on map type categories based on graphic representations, and yet, this had been an important issue in expert system development. In this project, studies were carried out on this aspect. Based on the analysis of the inherent characteristics of graphic map design, a hierarchical map type category was defined. The map type in this category emphasizes the graphic representations and is related to both data properties and map symbols. Therefore, map type can be an important link between data properties and map properties, and hence, can easily be applied to expert systems for thematic map design.

3. *The control scheme of map capacity.* To control map capacity in a fully automated mapping environment is a very difficult task. A commonly used method is to control the number of variables to be represented on the map by using a conservative fixed number. As a result, the system may be able to support only very simple mapping requirements. The schemes used in this project are based on the grouping of variables into layers; then, the maximum layer number is defined for each specific mapping requirement, and is adjusted according to the symbol types applied to each layer. In this way, the map can be safeguarded from overcrowding, and will not be too simple.

4. *The full explanation facility.* Expert systems developed in the field of cartography have been criticized because they failed to qualify as true expert systems due to the absence of an explanation facility. Unlike some of the systems, whose explanations are based on some "ready made documents" (Wang, 1992: 132), the explanation subsystem developed in TMDAS is directly related to the solution search, so that it is able to display the entire solution path, and the user can evaluate the reliability of the result from the solution path.

Based on the research performed so far, the following conclusions can be drawn:

The research and development of an expert system in thematic map design is necessary and feasible. Because maps have been used more and more with the use of GIS and electronic mapping systems, the function-rich and knowledge-absent status of cartographic components in software must be improved to prevent users from making and using misleading maps. Expert system technology provides an optimal solution to insert knowledge into the automated mapping environment.

The development of a fully operational expert system in cartography is a long term engineering project, and the concept design is one of the key issues towards final success. The heuristic characteristics of cartography make it a good candidate for expert system technology. Yet, the development of such a system should be considered as a long term engineering project. In this long term project, the importance of the concept design at different levels should be emphasized in order to guarantee that all subsystems and modules can eventually be integrated into one operational system.

The cartographic function and data property oriented approach is feasible in expert systems for thematic map design. This approach has been applied to the TMDAS prototype system with positive results. The approach makes the system able to map the subject domain independently – this is the advantage for cartographic expert systems.

Knowledge acquisition is the key issue in the development of an expert system in thematic map design, and much more effort is needed in this area. The biggest bottleneck in the development of a thematic map design system remains knowledge acquisition. It is hoped that research of cartographic expert systems and the study of cartographic theory can stimulate one another, and promote the development of both expert systems in cartography and of the cartographic domain itself.

Prototype implementation is necessary in system development. It is recommended to develop the initial prototype at an early stage in order to test the feasibility of the concept design, the efficiency of knowledge representation, and the validation of the formalized knowledge. The prototype is helpful because it can indicate the modifications needed at early stage and thus reduce costs.

TMDAS, after further testing and refinement, can be used as a cartographic advisor in certain environments such as education, research, or production. For this purpose, further testing and refinement of both the knowledge base and the user interface are needed. Complete documentation also needs to be prepared.

8.2 Suggestions for Further Development

The prototype of TMDAS is still in its infancy. A great deal of research effort mainly in the following three areas is required to achieve a fully operational system.

Intensive testing from both cartographic experts and end-user. Although some preliminary tests were done by the author, more tests are required by both cartographic experts and non-cartographic end-users, in order to test the system from different perspectives. Such tests include:

1. The validation of the knowledge and the completeness of the knowledge base;
2. The convenience and comprehensiveness of the user interface; and
3. The modification and enlargement of the knowledge base.

Based on feedback from the tests, both the knowledge base and the user interface may require modification.

Further extension of the knowledge base. The TMDAS knowledge base covers many aspects of thematic map design, but there are still many others to be incorporated in the

developed modules, and there are also new modules to be developed; some elements to be incorporated are listed below:

1. Further research into identification of visual variables;
2. Further research of the relationships between variables;
3. Research of factors which influence map design, from map use perspectives, such as the specific map objectives, and the different types of map readers;
4. New techniques for screen based maps and their perception properties;
5. Automatic extraction of data properties from GIS data bases;
6. Automatic or semi-automatic actual symbol design;
7. Other aspects of thematic map design such as base map selection, projection selection, legend design, map layout design, etc.

In the preceding list, some of the knowledge can be extracted, some of them may involve extremely intensive research, or some will rely on the development of related domains.

Integrating TMDAS with GIS/EMS. The prototype of TMDAS is a stand alone system. To make the system really useful, integration with GIS/EMS is needed. Technical issues related to this integration should be studied first, and this topic has been receiving more and more research attention. The problem may become simpler to solve with the use of new advanced data structures, such as the object-oriented approach or intelligent data bases, applied to the new generation of GIS. It will be necessary to select a proper GIS/EMS for the integration of TMDAS.

The research in this project and the development of the TMDAS prototype system have proven that expert system technology can be applied to the map design domain in order to introduce cartographic knowledge into the automated mapping environment. Current

research is still in its infancy in respect to both knowledge acquisition and system implementation; hence, further research is necessary.

BIBLIOGRAPHY

Adeli H. (1990) *Knowledge Engineering, Vol. 2, Applications*, McGraw-Hill Publishing Company, New York.

Armstrong M. P. (1991) "Knowledge classification and organization", In Buttenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 86-102.

Beard M. K. (1991) "Constraints on rule formation", In Buttenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 121-35.

Bertin J. (1983) *Semiology of Graphics*, University of Wisconsin Press, Madison, Wisconsin.

Bertin J. (1977) *Graphic and Graphic Information Processing*, Walter de Gruyter & Co., Berlin.

Biondo S. J. (1990) *Fundamentals of Expert Systems Technology, Principles and Concepts*, Ablex Publishing Corporation, New Jersey.

Bos E. S. (1984) "Systematic Symbol Design in Cartographic Education", *ITC Journal*, vol. 1, pp. 20-8.

Bossler J. D., D. L. Pendleton, G. F. Swetnam, R. L. Vitalo, C. R. Schwarz, S. Alper, and H. P. Danley (1988) "Knowledge-Based Cartography: The NOS Experience", *The American Cartographer*, Vol. 15, pp. 149-161.

Bouille F. (1984) "A Structured Expert System for Cartography Based on the HBDS", *Auto-Carto 6*, Ottawa, pp. 202-10.

Braßel K. E. and R. Weibel (1988) "A review and conceptual framework of automated map generalization", *International Journal of Geographical Information Systems*, Vol. 2, No. 3, pp. 229-244.

Brassel K. E. (1985) "Strategies and data models for computer-aided generalization", *International Yearbook of Cartography*, Vol. 25, pp. 11-29.

Buttenfield B. P. and D. M. Mark (1991) "Expert Systems in Cartographic Design", In Taylor, D. R. F. (ed), *Geographic Information System: The Computer and Contemporary Cartography*, Pergamon Press, Oxford, pp. 129-150.

Buttenfield B. P. and W. A. Mackaness (1991) "Visualisation", In Maguire D. J., M. F. Goodchild and D. W. Rhind, (eds.), Vol. 1, *Geographic Information Systems*, Longman Scientific & Technical Ltd., New York, pp. 427-443.

Castillo E. and E. Alvarez (1991) *Expert Systems: Uncertainty and Learning*, Computational Mechanics Publications, London.

Cook A. C. and C. B. Jones (1990) "A Prolog Interface to a Cartographic Database for Name Placement", *Proceedings of the 4th International Symposium on Spatial Data Handling*, Zurich, Vol. 1, pp. 701-710.

Covington M. A., D. Nute and A. Vellino (1988) *Prolog Programming in Depth*, Scott, Foresman and Company, London.

Cuff D. J. and M. T. Mattson (1982) *Thematic Maps, Their Design and Production*, Methuen & Co., New York.

Darwish A. M. and A. K. Jain (1988) "A Rule Based Approach for Visual Pattern Inspection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 1, pp. 56-68.

Dent B. D. (1990) *Cartography Thematic Map Design (2nd edition)*, Wm. C. Brown Publishers, U. S. A.

Egenhofer M. J. and A. U. Frank (1990) "Lobster: Combining AI and Database Techniques for GIS", *Photogrammetric Engineering and Remote Sensing*, Vol. 56, No. 6, pp. 919-926.

Forrest D. and A. Pearson (1990) "Map Design", In Perkins C. R. and R. B. Parry (eds.), *Information Sources in Cartography*, Bowker-Saur Ltd., London, pp. 169-188.

Fosher P. F. and W. A. Mackness (1987) "Are Cartographic Expert Systems Possible?" *Proceedings of AUTO Carto 8*, Baltimore, pp. 530-534.

Freeman H. and J. Ahn (1984) "AUTONAP - An Expert System for Automatic Map Name Placement", *Proceedings of 1st International Symposium on Spatial Data Handling*, Zurich, Vol. 2, pp. 544-571.

Golumbic M. C. (1990) *Advances in Artificial Intelligence -- Natural Language and Knowledge-based Systems*, Springer-Verlag, New York.

Goodchild M. F. (1990) "Geographic Information Systems and Cartography", *Cartography*, Vol. 19, No. 1, pp. 1-13.

Goodchild M. F. (1990) "Keynote Address, Spatial Information Science", *Proceedings of the 4th International Symposium on Spatial Data Handling*, Zurich, Vol. 1, pp. 3-12.

Gooding K. and D. Forrest (1991) "An examination of the difference between the interpretation of screen based and printed maps", *The Cartographic Journal*, Vol. 27, No. 1, pp. 15-19.

Gruber T. R. (1989) "The Acquisition of Strategic Knowledge", *Perspectives in Artificial Intelligence*, Vol. 4, Academic Press, Inc., Toronto.

Hart A. (1989) *Knowledge Acquisition for Expert Systems*, Kogan Page, London.

Heivly C. G. (1986) "Using Expert Systems Concepts to Fix USGS Digital Boundaries", *Proceedings of 2nd International Symposium on Spatial Data Handling*, Seattle, pp. 572-624.

Hua Y. and J. Gao (1993) "The establishment of a thematic map design expert system PC-mapper", *16th International Cartographic Conference*, pp. 943-947.

Jong W. M. and F. J. M. Wel (1990) "Embedded Artificial Intelligence and Spatial Data Handling, Some Research and Prototyping Experience", *Proceedings of the 4th International Symposium on Spatial Data Handling*, Zurich, Vol. 1, pp. 723-731.

Keates J. S. (1982) *Understanding Maps*, John Wiley, New York.

Kidd A. (ed.). (1987) *Knowledge Acquisition for Expert Systems*, Plenum Press, New York.

Kulhavy R. W., D. R. Pridemore and W. A. Stock (1992) "Cartographic Experience and Thinking Aloud about Thematic Maps", *Cartographica*, Vol. 29, No. 1, pp. 1-9.

Laurema M., O. Jaakkola, T. Sarjakoski and L. Schylberg (1991) "Knowledge Based Query Systems for Cartographic Visualization", *Reports of The Finnish Geodetic Institute*, pp. 2-40.

Lucas P. and L. V. D. Gaag (1991) *Principles of Expert Systems*, Addison-Wesley Publishing Company, Inc., Wokingham, England.

Luger G. F. (1992) "Applying Artificial Intelligence: Expert Systems, Neural Network, and Object-Oriented Design", *Course Materials*, Learning Tree International, Ottawa.

Luger G. F. and W. A. Stubblefield (1989) *Artificial Intelligence and the Design of Expert Systems*, Benjamin/Cummings Publishing Company, Inc., New York.

MacEachren A. M., B. P. Battenfield, J. B. Campbell, D. W. Dibiase and M. Monmonier (1992) "Visualization", In Abler, R. F., M. G. Marcus, and J. M. Olson (eds.), *Geography's Inner Worlds*, , Rutgers University Press, New Jersey, pp. 109-137.

Mark D. M. (1991) "Object Modelling and Phenomenon-based Generalization", In Battenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 103-118.

McMaster R. B. (1991) "Conceptual frameworks for geographical knowledge", In Battenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 21-39.

McGraw K. L. and C. R. Westphal (1990) *Readings in Knowledge Acquisition -- Current Practices and Trends*, Ellis Horwood, New York.

McTear M. F. and T. J. Anderson (1990) *Understanding Knowledge Engineering*, Ellis Horwood Limited, New York.

Medyckyj-scott D. and C. Board (1991) "Cognitive Cartography: A New Heart for a Lost Soul", in Muller (eds), *Advances in Cartography*, Elsevier Science Publishers Ltd., London, pp. 201-230.

Molenaar M. (1989) "Towards a Geographic Information Theory", *ITC Journal*, Vol. 1, pp. 5-11.

Morrison J. L. (1974) "A theoretical framework for cartographic generalization with emphasis on the process of symbolization", *International Yearbook of Cartography*, pp. 115-127.

Muehrcke P. C. (1986) *Map Use*, JP Publications, Madison, WI.

Muggleton S. (1990) *Inductive Acquisition of Expert Knowledge*, Turing Institute Press in association with Addison-Wesley Publishing Company, New York.

Mulder J. A., A. K. Mackworth and W. S. Havens (1988) "Knowledge Structuring and Constraint Satisfaction: The Mapsee Approach", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 10, No.6, pp. 866-879.

Muller J. C. (1991) "The Cartographic Agenda of the 1990s: Updates and Prospects", *ITC Journal*, Vol. 2, pp. 55-62.

Muller J. C. and Z. S. Wang (1990) "A Knowledge Based System for Cartographic Symbol Design", *The Cartographic Journal*, Vol. 27, No. 1, pp. 24-30.

Muller J. C., R. D. Johnson and L. R. Vanzella (1986) "A Knowledge-based Approach for Developing Cartographic Expertise", *Proceedings of the 2nd Symposium on Spatial Data Handling*, Seattle, pp. 557-571.

Nickerson B. G. and H. Freeman (1986) "Development of a rule-based system for automatic map generalization", *Proceedings, Second International Symposium on Spatial Data Handling*, Seattle, pp. 537-556.

Niemann H., G. F. Sagerer, S. Schroder and F. Kummert (1990) "ETNEST: A Semantic Network System for Pattern Understanding", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 9, pp. 883-905.

Nyerges T. L. (1991) "Representing geographical meaning", In Battenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 59-85.

Nyerges T. and P. Jankowski (1989) "Design Considerations for MAPKBS-MAP Projection Knowledge-based System", *The American Cartographer*, Vol. 16, No. 2, pp. 85-95.

Payne E. C. and R. C. McArthur (1990) *Developing Expert Systems - A Knowledge Engineer's Handbook for Rules and Objects*, John Wiley & Sons Inc., New York.

Pfefferkorn C., D. Burr, D. Harrison and B. Heckman (1985) "ACES: A Cartographic Expert System", *Proceedings of Auto Carto 7*, pp. 399-407.

Parsay K., M. C. S. Khoshafian and H. Wong (1990) "Intelligent Databases", *AI Expert*, Vol. 3, pp. 38-47.

Partridge D. (1991) *A New Guide to Artificial Intelligence*, Ablex Publishing Corporation, New Jersey.

Patterson D. W. (1990) *Introduction to Artificial Intelligence and Expert Systems*, Prentice-Hall, Inc., New Jersey.

3 of/de 3

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

Rase W. D. (1991) "Computer-assisted Map Design", In Muller, J. C. (ed), *Advances in Cartography*, Elsevier Science Publishers Ltd., London, pp. 181-200.

Rhind D. (1988) "A GIS Research Agenda", *International Journal of Geographical Information Systems*, Vol. 2, No. 1, pp. 23-28.

Richardson D. E. and J. C. Muller R. (1991) "Rule selection for small-scale map generalization", In Buttenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation* Longman, London, pp. 136-149.

Robertson P. K. (1988) "Choosing Data Representations for The Effective Visualisation of Spatial Data", *Proceedings of The Third International Symposium on Spatial Data Handling*, Sydney, Australia.

Robinson A. R. Sale, J. Morrison and P. C. Muehrcke (1984) *Elements of Cartography* (5th edition), John Wiley & Sons Inc., New York.

Robinson G. and M. Jackson (1985) "Expert Systems in Map Design", *Proceedings of Auto Carto 7*, New York, pp. 510-517.

Robinson V. B. and A. U. Frank (1987) "Expert Systems for Geographic Information systems", *PE & RS*, Vol. 111, No. 10, pp. 1435-1441

Robinson V. B. and A. U. Frank (1987) "Expert Systems Applied to Problems in Geographic Information Systems: Introduction, Review and Prospect", *Proceedings of AUTO Carto 8*, Baltimore, pp. 510-517.

Schutzer D. (1987) *Artificial Intelligence -- An Applications-oriented Approach*, Van Nostrand Reinhold Company, New York.

Shea K. S. (1991) "Design considerations for an artificially intelligent system", In Buttenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation*, Longman, London, pp. 3-20.

Shiryaev E. E. (1987) *Computers and the representation of geographical data*, John Wiley & Sons, New York.

Siekierska E. M. (1983) "Towards an Electronic Atlas", *Proceedings of International Symposium on Computer-Assisted Cartography, AUTO Carto 6*, Ottawa, pp. 464-474

Siekierska E. M. and D. R. F. Taylor (1991) "Electronic Mapping and Electronic Atlases: New Cartographic Products for the Information Era - the Electronic Atlas of Canada", *CISM Journal*, Vol. 45, pp. 11-21.

Slocum T. A. and S. L. Egbert (1991) "Cartographic Data Display", In Taylor D. R. F. (ed), *Geographic Information System: The Computer and Contemporary Cartography*, Pergamon Press, Oxford, pp. 167-200.

Smith R. (1987) "Electronic Atlas of Arkansas: Design and Operational Considerations", *Proceedings of the 13th International Cartographic Conference*, Morelia, Mexico, pp. 161-166.

Smith T. R., D. Peuquet, S. Menon and P. Agarwal (1987) "KBGIS II: A Knowledge-based Geographic Information System", *International Journal of Geographical Information System*, Vol. 1, No. 2, pp.149-172.

Smith T. R. and Yiang J. (1991) "Knowledge-Based Approachs in GIS", In Maguire D. J., M. F. Goodchild and D. W. Rhind, (eds.), Vol. 1, *Geographic Information Systems*, Longman Scientific & Technical Ltd., New York, pp. 413-425.

Su B., W. Zhang, H. Li, Y. Zhang, X. Zhang, X. Zhu and J. Li (1993) "A knowledge base system based on GIS for thematic mapping", *16th International Cartographic Conference*, pp. 468-477.

Taylor D. R. F. (1987) "New Methods and Technologies in Cartography: The Digital World Atlas", *International Yearbook of Cartography*, pp. 219-224.

Taylor D. R. F. (1991) "A conceptual basis for cartography: new directions for the information era", *The Cartographic Journal*, Vol. 28, pp. 213-216.

Taylor D. R. F. (1987) "Cartographic communication on computer screens: the effect of sequential presentation of map information", *Proceedings of the 13th International Cartographic Conference*, Vol. 1, Morelia, Mexico, pp. 591-611.

Tomlin C. D. (1991) "Cartographic modelling", In Maguire, D. J., M. F. Goodchild and D. W. Rhind, (eds.), Vol. 1, *Geographic Information Systems*, Longman Scientific & Technical Ltd., New York, pp.361-374.

Thompson J. J. S. (1989) "The National Atlas Information System: An Overview", *Proceedings of Canadian Conference Challenges for 1990's GIS*, CISM, Ottawa, Canada, pp. 62-74.

Tufte E. R. (1983) *The Visual Display of Quantitative Information*, Graphics Press, Cheshire, Connecticut.

Tufte E. R. (1990) *Envisioning Information*, Graphics Press, Cheshire, Connecticut.

Upton C., T. Jr. Faulhaber, D. Kamins, D. Laidlaw, D. Schlegel, J. Vroom, R. Guewitz and A. V. Dam (1989) "The Application Visualization System: A Computational Environment for Scientific Visualization", *IEEE Computer Graphics & Applications*, pp. 30-42.

Wang J. Y. F. Wu and Z. Wu (1993) "The research on tools of the cartographic generalization expert system", *16th International Cartographic Conference*, pp. 221-227.

Wang Z. (1990) "A Representation Schema for Cartographic Information", *Proceedings of the 4th International Symposium on Spatial Data Handling*, Zurich, Vol. 1, pp. 782-791.

Wang Z. (1992) *An Expert System for Cartographic Symbol Design*, published Ph.D. dissertation, ITC Enschede.

Weibel R. (1991) "Amplified intelligence and rule-based system", In Battenfield B. P. and R. B. McMaster (eds.), *Map Generalization: Making Rules for Knowledge Representation* Longman, London, pp. 172-186.

Weiskamp K. and T. Hengl (1988) *Artificial Intelligence Programming with Turbo Prolog*, John Wiley & Sons, Inc., New York.

Williams R. J. (1987) "Evolution in cartography: data intelligence", *Cartography*, Vol. 16, No. 2, pp.141-146.

Worth C. (1989) "Problems with experimental research in map design - a case study", *The Cartographic Journal*, Vol. 26, pp. 148-153.

Yang J., E. M. Siekierska and D. R. F. Taylor (1993) "Concept design and knowledge acquisition for the thematic map design advisory system (TMDAS)", *Proceedings, the Canadian Conference on GIS*, Ottawa, pp. 474-484.

Zhang K. and R. Huang (1980) *Design and Compilation of Thematic Maps*, Publishing House of Surveying and Mapping, Beijing (in Chinese).

APPENDIX

Demonstrations of the System Execution of Case Tests

C > TMDAS

J.shell is an expert system shell developed in Turbo Prolog. J.shell employs a backward-chaining inference engine. Any knowledge base built in the format of J.shell can use J.shell as the inference engine.

J.shell format:

fact(fact_number,fact)

rule(rule_number,then,ifs)

question(question_number,question)

explanation(explanation_number,question,explanation)

utility(utility_number,utility)

validresponse(validre_number,question,response,check_response,converted_response)

In the preceding expressions, the symbols used before the parentheses are the names of the predicates. The symbols between the parentheses and separated by commas indicate the number of elements in the predicate and their data types. In J.shell, the types are defined as follows:

fact_number, rule_number, question_number, explanation_number, utilities_number,
validre_number: INTEGER
explanation, response: SYMBOL
fact, then, question, utilities, check_response, converted_response: DATA_TYPE
ifs: DATA_LIST

where DATA_TYPE and DATA_LIST are user defined types. DATA_TYPE includes all the data types used in the system, and DATA_LIST defines a list whose elements have the types defined by DATA_TYPE.

Explanation facilities of the expert system, WHY", "HOW", and "Explain", have been implemented in the system.

For further information on J.shell, refer to J.shell introduction in the thesis.

TMDAS stands for Thematic Map Design Advisory System. It is a knowledge base developed in the J.shell format to give advice on thematic map design. TMDAS aims to insert knowledge of map design into the automated mapping environment; it can also be used as cartographic advisor for general purposes.

The data to be mapped can come from two types of sources: the NAIS (National Atlas Information System of Canada) data set, or a user-brought data set. These two types of data can be used separately or be combined. If NAIS data is used, the user can select the variable name from the list, while if user-brought data is used, the user is expected to answer some questions related to data properties.

When the system asks questions, the user can either answer back, or ask for either further explanations of the question, or the reason why the question is being asked. The "Explanation" and "WHY" functions will give the required information. If the user answers back, the system will further check the validity of the answer.

After obtaining all the needed information, the system will give a recommendation for map design, based on its reasoning. The recommendation includes the number of layers grouped for the map, the number of variables and the variable names in each layer, the map types selected for each layer, and most importantly, the symbol types, visual variables and visual constants selected for each layer.

Having obtained the recommendation, the user can: 1) SAVE the recommendation as an ASCII file which can be printed; 2) VIEW the example maps corresponding to the recommendation; 3) ask HOW the recommendation was reached to see the solution path by BROWSE, displaying the whole solution path at once, or DYNAMIC, displaying the path node by node; or, 4) CONTINUE the process to make another consultation or to quit the system.

For further information on TMDAS knowledge, refer to the TMDAS introduction in the thesis.

[Each of the three windows has its own functions. When “Explain” [e] / “Why” [w] is asked, “Explanation” window / “Why” window will become the active window. At other times, the “Dialog” window is the active one. The information recorded for each case test is displayed in the “Dialog” window].

Case 1: CANADA LAND COVER ASSOCIATIONS

variable(1, land_cover)

spatial_dimension(land_cover, total_area)

measurement_scale(land_cover, nominal)

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;

2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;

2. other;

3. both;

Enter your choice: 1

SELECTED NAIS DATA SET

1. Native_population	8. Pump_stations	15. Residual_areas
2. Native_linguistic	9. Oil_associated_fields	16. Urban_service
3. Native_census_area	10. Airports	17. Residual_service
4. Indian_population	11. Longest_day	18. Urban_population
5. Inuit_population	12. Shortest_day	19. Residual_population
6. Land_cover	13. Climate_station	20. Days_above_5_degree
7. Oil_pipelines	14. Urban_employment	

Enter each choice followed by return [0 for exit]: 6

Enter each choice followed by return [0 for exit]: 0

You have selected:

1. Land_cover

Are they correct [y/n [y]]: y

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
---------------	------------------	------------------------	------------------------	------------------------

1	Land_cover			
1	1	area/ category	hue form	value size, value, spacing

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

Case 2: CANADA FARM OPERATORS

variable(1, farm_operators)

spatial_dimension(farm_operators, discrete_area)

measurement_scale(farm_operators, absolute_quantitative)

variable(2, farm_land_change)

statistical_map(farm_land_change, yes)

measurement_scale(farm_land_change, derived_quantitative)

variable(3, agricultural_activity)

spatial_dimension(agricultural_activity, point)

measurement_scale(agricultural_activity, nominal)

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;

2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;
2. other;
3. both;

Enter your choice: 2

Enter the number of variables [<13]: 3

Enter the name of variable 1: farm_operators

Is the variable: farm_operators
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
farm_operators

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 3

What is the measurement scale of the variable:
farm_operators

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 2: farm_land_change

Is the variable: farm_land_change
aggregated by unit area [y/n]? y

What is the measurement scale of the variable:
farm_land_change

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 4

Enter the name of variable 3: agricultural_activity

Is the variable: agricultural_activity
aggregated by unit area [y/n]? n

Enter your choice: 2

The map is designed as a:

1. simple map;
2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;
2. other;
3. both;

Enter your choice: 2

Enter the number of variables [<13]: 3

Enter the name of variable 1: lumber_production

Is the variable: lumber_production
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:

lumber_production

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 1

What is the measurement scale of the variable:

lumber_production

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 2: wood_type

Is the variable: wood_type
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:

wood_type

1. point
2. line
3. discrete area

4. partial area
5. total area
6. volume

Enter your choice: 1

What is the measurement scale of the variable:

wood_type

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 1

Enter the name of variable 3: forest_area

Is the variable: forest_area

aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:

forest_area

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 4

What is the measurement scale of the variable:

forest_area

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 1

What is the relationship between

lumber_production and wood_type

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between

wood_type and forest_area

1. consist_of
2. part_of
3. neighbour_of
4. same_location

5. related_to
Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Lumber_production 1	point/ point	size	form, hue
1	Wood_type 1	point/ point	hue	value, size
2	Forest_area 1	area/ area	hue form	value size, value, spacing

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: v

Layer#: 2 Xth recommendation: 2

[Map will appear on the screen]

Save/View/How/Continue: c

Case 4: CANADA OIL PIPELINES

```
variable(1, oil_pipelines)
spatial_dimension(oil_pipelines, line)
measurement_scale(oil_pipelines, nominal)

variable(2, pump_stations)
```

```

spatial_dimension(pump_stations,point)

measurement_scale(pump_stations, nominal)

variable(3, oil_fields)

spatial_dimension(oil_fields, p_area)

measurement_scale(oil_fields, nominal)

```

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;

2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;

2. other;

3. both;

Enter your choice: 1

SELECTED NAIS DATA SET

1. Native_population	8. Pump_stations	15. Residual_areas
2. Native_linguistic	9. Oil_associated_fields	16. Urban_service
3. Native_census_area	10. Airports	17. Residual_service
4. Indian_population	11. Longest_day	18. Urban_population
5. Inuit_population	12. Shortest_day	19. Residual_population
6. Land_cover	13. Climate_station	20. Days_above_5_degree
7. Oil_pipelines	14. Urban_employment	

Enter each choice followed by return [0 for exit]: 7

Enter each choice followed by return [0 for exit]: 8

Enter each choice followed by return [0 for exit]: 9

Enter each choice followed by return [0 for exit]: 0

You have selected:

1. Oil_pipelines

2. Pump_stations

3. Oil_fields

Are they correct [y/n [y]]: y

What is the relationship between
oil_pipelines and pump_stations

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
oil_pipelines and oil_fields

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
pump_stations and oil_fields

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Oil_pipelines 1	line/ line	hue	value, size
2	Pump_stations 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue
3	Oil_fields 1	area/ area	hue form	value size, value, spacing

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

Case 5: CANADA POPULATION DISTRIBUTION 1976

variable(1, place_population)

spatial_dimension(place_population, point)

measurement_scale(place_population, absolute_quantitative)

variable(2, discrete_population)

statistical_map(discrete_population, yes)

measurement_scale(discrete_population, absolute_quantitative)

variable(3, isolated_settlement)

spatial_dimension(isolated_settlement, point)

measurement_scale(isolated_settlement, nominal)

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;

2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;

2. other;

3. both;

Enter your choice: 2

Enter the number of variables [<13]: 3

Enter the name of variable 1: place_population

Is the variable: place_population
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
place_population

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 1

What is the measurement scale of the variable:
place_population

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 2: discrete_population

Is the variable: discrete_population
aggregated by unit area [y/n]? y

What is the measurement scale of the variable:
discrete_population

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 3: isolated_settlement

Is the variable: isolated_settlement
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
isolated_settlement

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 1

What is the measurement scale of the variable:

isolated_settlement

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 1

What is the relationship between

place_population and discrete_population

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between

place_population and isolated_settlement

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between

discrete_population and isolated_settlement

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Place_population 1	point/ point	size	form, hue
2	Discrete_population 1	dot/ dot	*	form, hue
3	Isolated_settlement 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: v

Layer#: 2 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

Case 6: CANADA SOLAR RADIATION-ANNUAL

variable(1, climate_station)

spatial_dimension(climate_station, point)

measurement_scale(climate_station, nominal)

variable(2, solar_radiation)

spatial_dimension(solar_radiation, volume)

measurement_scale(solar_radiation, derived_quantitative)

variable(3, radiation_variability)

spatial_dimension(radiation_variability, volume)

measurement_scale(radiation_variability, derived_quantitative)

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;
2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;
2. other;
3. both;

Enter your choice: 3

SELECTED NAIS DATA SET

1. Native_population	8. Pump_stations	15. Residual_areas
2. Native_linguistic	9. Oil_associated_fields	16. Urban_service
3. Native_census_area	10. Airports	17. Residual_service
4. Indian_population	11. Longest_day	18. Urban_population
5. Inuit_population	12. Shortest_day	19. Residual_population
6. Land_cover	13. Climate_station	20. Days_above_5_degree
7. Oil_pipelines	14. Urban_employment	

Enter each choice followed by return [0 for exit]: 13

Enter each choice followed by return [0 for exit]: 0

You have selected:

1. Climate_station

Are they correct [y/n [y]]: y

Enter the number of extra variables [<13]: 2

Enter the name of variable 2: solar_radiation

Is the variable: solar_radiation
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
solar_radiation

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 6

What is the measurement scale of the variable:
solar_radiation

1. nominal
2. ordinal

3. absolute quantitative
4. derived quantitative

Enter your choice: 4

Does the variable: solar_radiation
represent periodically changed data [y/n]? n

Enter the name of variable 3: radiation_variability

Is the variable: radiation_variability
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
radiation_variability

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 6

What is the measurement scale of the variable:
radiation_variability

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 4

Does the variable: radiation_variability
represent periodically changed data [y/n]? n

What is the relationship between
climate_station and solar_radiation

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
climate_station and radiation_variability

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
solar_radiation and radiation_variability

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	climate_station 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue
2	Solar_radiatio 1	isoline/ isoline	*	form, hue
3	radiation_variability 1	isoline/ isoline	*	form, hue

Save/View/How/Continue: v

Layer#: 3 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

Case 7: CANADA MONTHLY DISTRIBUTION OF RUNOFF

variable(1, jan_runoff)

spatial_dimension(jan_runoff, volume)

measurement_scale(jan_runoff, derived_quantitative)

periodically_changed(jan_runoff,yes)

variable(2, feb_runoff)

spatial_dimension(feb_runoff, volume)
measurement_scale(feb_runoff, derived_quantitative)
periodically_changed(feb_runoff, yes)

The final map will be on:

1. screen;
2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;
2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;
2. other;
3. both;

Enter your choice: 2

Enter the number of variables [<13]: 12

Enter the name of variable 1: jan_runoff

Is the variable: jan_runoff
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
jan_runoff

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 6

What is the measurement scale of the variable:
jan_runoff

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 4

What is the spatial dimension of the variable:
agricultural_activity

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 1

What is the measurement scale of the variable:
agricultural_activity

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 1

What is the relationship between
farm_operators and farm_land_change

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
farm_operators and agricultural_activity

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

What is the relationship between
farm_land_change and agricultural_activity

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
---------------	------------------	------------------------	------------------------	------------------------

Farm_operators

Does the variable: jan_runoff
represent periodically changed data [y/n]? y

Enter the name of variable 2: feb_runoff

Is the variable: feb_runoff
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:
feb_runoff

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 6

What is the measurement scale of the variable:
feb_runoff

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 4

Does the variable: feb_runoff
represent periodically changed data [y/n]? y

[The procedure is repeated for another ten months]

What is the relationship between
jan_runoff and feb_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and mar_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and apr_runoff

1. consist_of

1	1	dot/ dot	*	form, hue
2	Farm_land_change 1	area/ choropleth	spacing value hue_value	form, hue, size hue
3	Agricultural_activity 1	point/ point	hue form orientation	value, size size, value, hue size, value, form, hue

Save/View/How/Continue: v

Layer#: 2 Xth recommendation: 2

[Map will appear on the screen]

Save/View/How/Continue: c

Case 3: CANADA SAWMILLS

variable(1, lumber_production)

spatial_dimension(lumber_production, point)

measurement_scale(lumber_production, absolute_quantitative)

variable(2, wood_type)

spatial_dimension(wood_type, point)

measurement_scale(wood_type, nominal)

variable(3, forest_area)

spatial_dimension(forest_area, partial_area)

measurement_scale(forest_area, nominal)

The final map will be on:

1. screen;
2. paper;

2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and may_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and jun_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and jul_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and aug_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and sep_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and oct_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and nov_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

What is the relationship between
jan_runoff and dec_runoff

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 4

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
1	Dec_runoff Nov_runoff Oct_runoff Sep_runoff Aug_runoff Jul_runoff Jun_runoff May_runoff Apr_runoff Mar_runoff Feb_runoff Jan_runoff	special_graphics/ located_graphics	size	form, hue

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

Case 8: CANADA THE AGING POPULATION

variable(1, total_elder)

statistical_map(total_elder, yes)

measurement_scale(total_elder, absolute_quantitative)

variable(2, 65_group)

statistical_map(65_group, yes)

measurement_scale(65_group, absolute_quantitative)

variable(3, 75_group)

statistical_map(75_group, yes)

measurement_scale(75_group, absolute_quantitative)

variable(4, census_boundary)

spatial_dimension(census_boundary, line)

measurement_scale(census_boundary, nominal)

The final map will be on:

1. screen;

2. paper;

Enter your choice: 2

The map is designed as a:

1. simple map;

2. complex map;

Enter your choice: 2

Define the data used:

1. NAIS;

2. other;

3. both;
Enter your choice: 2

Enter the number of variables [<13]: 4

Enter the name of variable 1: total_elder

Is the variable: total_elder
aggregated by unit area [y/n]? y

What is the measurement scale of the variable:
total_elder

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 2: 65_group

Is the variable: 65_group
aggregated by unit area [y/n]? y

What is the measurement scale of the variable:
65_group

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 3: 75_group

Is the variable: 75_group
aggregated by unit area [y/n]? y

What is the measurement scale of the variable:
75_group

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 3

Enter the name of variable 4: census_boundary

Is the variable: census_boundary
aggregated by unit area [y/n]? n

What is the spatial dimension of the variable:

census_boundary

1. point
2. line
3. discrete area
4. partial area
5. total area
6. volume

Enter your choice: 2

What is the measurement scale of the variable:

census_boundary

1. nominal
2. ordinal
3. absolute quantitative
4. derived quantitative

Enter your choice: 1

What is the relationship between

total_elder and 65_group

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 1

What is the relationship between

total_elder and 75_group

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 1

What is the relationship between

total_elder and census_boundary

1. consist_of
2. part_of
3. neighbour_of
4. same_location
5. related_to

Enter your choice: 5

RECOMMENDATION FROM TMDAS

<u>Layer#</u>	<u>Variables</u>	<u>Symbol/Map Type</u>	<u>Visual Variable</u>	<u>Visual Constant</u>
---------------	------------------	------------------------	------------------------	------------------------

75_group				
----------	--	--	--	--

	65_group			
	Total_elder			
1	3	compound_point/ area_based_point	size	form, hue
2	Census_boundary			
	1	line/ line	hue	value, size

Save/View/How/Continue: v

Layer#: 1 Xth recommendation: 1

[Map will appear on the screen]

Save/View/How/Continue: c

r -- Run, q -- Quit
Enter option (r/q): q

END

2 1-0 7-9 4

FIN