COMPUTER AUTOMATION OF LAND EVALUATION PROCEDURES

Michael Pooley

Doctor of Philosophy

The University of Aston in Birmingham
July 1989

THE UNIVERSITY OF ASTON IN BIRMINGHAM

COMPUTER AUTOMATION OF LAND EVALUATION PROCEDURES

MICHAEL POOLEY. DOCTOR OF PHILOSOPHY, 1989.

SUMMARY

This study was concerned with the computer automation of land evaluation. This is a broad subject with many issues to be resolved, so the study concentrated on three key problems: knowledge based programming; the integration of spatial information from remote sensing and other sources; and the inclusion of socio-economic information into the land evaluation analysis. Land evaluation and land use planning were considered in the context of overseas projects in the developing world.

Knowledge based systems were found to provide significant advantages over conventional programming techniques for some aspects of the land evaluation process. Declarative languages, in particular Prolog, were ideally suited to integration of social information which changes with every situation. Rule-based expert system shells were also found to be suitable for this role, including knowledge acquisition at the interview stage. All the expert system shells examined suffered from very limited constraints to problem size, but new products now overcome this. Inductive expert system shells were useful as a guide to knowledge gaps and possible relationships, but the number of examples required was unrealistic for typical land use planning situations.

The accuracy of classified satellite imagery was significantly enhanced by integrating spatial information on soil distribution for Thailand data. Estimates of the rice producing area were substantially improved (30% change in area) by the addition of soil information. Image processing work on Mozambique showed that satellite remote sensing was a useful tool in stratifying vegetation cover at provincial level to identify key development areas, but it's full utility could not be realised on typical planning projects, without treatment as part of a complete spatial information system.

KEY WORDS : LAND USE PLANNING; LAND EVALUATION; IKBS; PROLOG; SATELLITE IMAGERY.

ACKNOWLEDGEMENTS

Thanks are due to Dr T.R.E. Chidley, Dr D. Van Rest, Dr J. Elgy, Dr W.G. Collins and Dr D. Moffatt for their supervision throughout this research project.

The support of colleagues at the remote sensing unit in Aston University and ULG Consultants Ltd was also appreciated. The author is grateful for the financial support received for this research from the Scientific and Engineering Research Council and ULG.

The cooperation of the Department of Land Development in Bangkok Thailand was also crucial to the completion of this study. The collaboration of all those involved was appreciated. Special thanks go to Khun Chakrit; Khun Pananya; Dr Samarn; Khun Awud; Khun Prattana; Khun Ariyaporn; Khun Wirat and Charlie.

CONTENTS

CHAPTER VII DISCUSSION AND CONCLUSION

LIST OF FIGURES

Figure Page

LIST OF PLATES

LIST OF TABLES

Table Page

LIST OF APPENDICES

ABBREVIATIONS

Because the subject matter of this thesis deals with disciplines dominated by jargon and acronyms, it has been impossible to avoid adopting some of the abbreviations and new additions to the English language. For this reason the list of abbreviations is provided here and a glossary follows after the references section. In discussing computer programs, to avoid confusion with other text, a convention of highlighting file names in bold, was also adopted.

AI Artificial Intelligence.

ASCII American Standard Code for Information Interchange. A character coding convention whereby a character is represented by a unique group of 7 bits. These 7 bits may be preceded by a parity bit which depends on the system in use.

DBMS Database Management System. A system which organises a database structure and provides access to information in it.

DIPS Digital Image Processing System

DLD Department of Land Development (Thailand)

FAO Food and Agricultural Organisation of the UN

FCC Fertility Capability Classification

GIS Geographic Information System

IKBS Intelligent Knowledge Based Systems

LECS Land Evaluation Computer System

LU Land Unit

LUT Land Utilisation Type (also an abbreviation for Look-Up Table in the context of a computer system)

MB Million baht

Mb Megabyte : One million bytes

Min. of Ag. Ministry of Agriculture in the UK.

MSS Landsat Multi-Spectral Scanner, one of the sensors aboard the American earth resources satellites. The scanner measures reflected radiation in 4 spectral channels at 80 m resolution.

OAE Office of Agricultural Extension, Thailand.

ORACLE A relational database management system

RTG Royal Thai Government.

S.S.E. & W. Soil Survey of England & Wales.

SQL Structured Query Language

TM Thematic Mapper, sensor aboard Landsat satellites 4 and 5. Measuring radiation in 7 channels at 30 m pixel resolution.

CHAPTER I

INTRODUCTION

1.1 Background and Objectives

This thesis is concerned with the application of Information Technology (IT) to the process of land evaluation and planning. By it's very nature, Information Technology is a diverse subject. The type of techniques vary with the application and both software and hardware developments may influence successful automation of an application. In recent years the IT revolution has become evident in everyday life. This has been largely due to the advancement in hardware, since the advent of the integrated circuit in 1959. From that time the performance of integrated circuits has improved by a factor of more than 10,000, whilst cost remained virtually unchanged (Meindl, 1987). This has meant the tools for automation have become more widely available for solving problems. However the solution also depends on software development, which ultimately determines the rate of transition to a society which can take full advantage of the microchip.

Land evaluation is also a diverse subject, of an interdisciplinary nature. In attempting to recommend the optimal land use for a particular area all major aspects and consequences of that land use must be considered. This entails assessment of several sectors including physical resources; socio-economic factors; environmental effects; macro economic consequences and political policies. For most land evaluation schemes in the past, concerned with planning of agricultural development, only the physical resources have been considered. Indeed in many developing countries, land capability has been synonymous with soil capability. However, in practice government departments with responsibility for land use planning in developing countries cannot ignore the broader considerations. In this sense land evaluation schemes are still very rudimentary. Yet implementation of land evaluation procedures is usually a complicated and subjective process. Often it is not successfully achieved, either in terms of arriving at justifiable recommendations, or providing a plan within a sensible time-scale. Without improvements in processing, further complication of evaluatior procedures, are not likely to be achieved.

The objective of this thesis is to show how recent advancements in IT can be applied, to improve the process of land use evaluation and planning. The intention is to show how through the application of IT, the current procedures could be automated and also to indicate how flexibility could be introduced and how the evaluation process can be improved, to incorporate information from a wider range of sectors.

The work was done in collaboration with a locally based overseas consultancy company (ULG of Warwick), specialising in agricultural projects and environmental studies.

With a broad topic to cover, it was considered essential to concentrate on separate components, as opposed to building a comprehensive system. Organisations such as the FAO have been working on a universally applicable land evaluation computer system for at least the last 10 years and have not arrived at a satisfactory solution yet (see section 2.2). At the inception of this work (1985) knowledge-based programming and particularly expert systems were much publicised as the solution to problems not previously considered as suitable for complete computer automation, because of their dependence on expert knowledge. The application of such systems to land evaluation, to determine how appropriate the systems are for the task was adopted as the first aim.

Spatial variability of data is inherent to the process of land evaluation and must be considered in order to achieve a dependable land use plan. Therefore a second aim was to show how spatially related data such as satellite imagery and digital maps, could be integrated in an automated land evaluation system, and determine whether this improved the evaluation.

The past emphasis of land evaluation based on physical resources alone has given an impression to some, that land evaluation is not pertinent to problems of land use policy facing governments, which have progressed beyond resource inventory. Crucial to the remedying of this situation is the incorporation of financial and socio-economic analysis to the procedure. To this end an investigation was made as to how IT techniques could be used to aid in the gathering and analysis of socio-economic information. This was the third aim of the study.

Although the three key components: knowledge based systems; spatial relationships; and socio-economic considerations, do not constitute a complete solution to the automation of land evaluation, they are key elements in accomplishing the tasks and the results should prove the feasibility of the approach.

Land evaluation in the context of optimisation of land use in developing countries is another factor considered in this work. For an application study it was considered essential to conduct the research in a practical environment, with the aim of solving a real problem rather than an illusory one. Thus throughout the study, work done is related to the implementation of land use planning in developing countries. Field studies were made in Thailand and Mozambique. Advancements in hardware, including cost/performance assessments, will also be made as their effect on the pertinence of an IT approach is substantial and results in a constant change in the feasibility of the approach.

1.2 Land Evaluation

Land evaluation is a procedure, by which the properties of land important to it's usage, are assessed to yield a measure of the utility of the land. This is a deliberately broad definition, because in practice land evaluation is undertaken for a wide variety of purposes and the nature and definition of an evaluation may change according to the ultimate intention. In all cases the objective is to estimate the value of the land, whether by subjective or objective means. Whichever land evaluation system is used, considerable experience and knowledge are required to arrive at a sensible land use plan.

According to Young (1976) the data employed in evaluation come from three main sources : natural resource survey; the technology of resource use; and economics. Young referred to economics in it's widest sense and this would be better described as socio-economic influences, since tenure and pattern of labour availability are some of the important factors considered under this sector. A dictionary definition of land is (Kirkpatrick 1983):

> "The solid portion of the surface of the globe."

but this is an inadequate definition on which to proceed to a discussion of land evaluation. The FAO (1976) definition is:

> "Land comprises the physical environment, including climate, relief, soils, hydrology and vegetation to the extent that these influence potential for land use. It includes the results of past and present human activity, e.g. reclamation from the sea, vegetation clearance, and also adverse results, e.g. soil salinization.

> Purely economic and social characteristics, however, are not included in the concept of land; these form part of the economic and social context."

In the same report the FAO defines a land mapping unit as " a mapped area of land with specified characteristics," Such areas are defined from the natural resource surveys. In the standard approach to land development planning, as practiced by most development agencies for the last 30 years or so, the process is regarded as consisting of three phases: description; evaluation; and development (Young, 1976). The description phase includes natural resource surveys such as soil survey and current land use practices. The technology of resource use, such as crop requirements and agricultural methods, is then combined with the information from resource survey, at the evaluation stage, to obtain a measure of potential for each of the uses considered. The development phase then considers the socio-economic context and a land use plan is derived, based on comparative potential of the land for different uses .

To some extent the three stages are sequential, with information collected at one phase being passed on as the basis for the next . When the three-phase approach is adopted at different scales of survey a cyclic pattern is evident across the levels of survey, as shown in table 1.1, taken from Young (1976). Young himself criticises the three-phase approach he identifies, citing the arguments of others to support his case.

Moss (1968, 1969) has argued for an alternative approach to resource survey than that centered on geomorphology; favouring a non sequential dynamic approach based on ecology. The essence of the criticism is that with a land systems type approach, based on geomorphology and soils, a static approach is engendered, because land form changes only very slowly. Thus, land tends to be considered as a finite exploitable resource. In reality the land use will affect the resources themselves. An example of this is the adoption of an intensive land use in an area traditionally used for shifting cultivation. Soil organic matter and nutrient levels will vary over the area according to the age of fallow. This is unlikely to be

TABLE 1.1 The place of land evaluation in an idealized sequence of development planning. Based in part on Brinkman and Smyth (1973).

Purpose of survey	Phase
Resource inventory /Project location	1. Resource Surveys, reconnaissance scale; including soil-landform or land system map.
	2. Land evaluation: qualitative suitability for major kinds of land use.
	3. Preparation of regional plan; identification of possible projects.
Feasibility survey of possible project(s)	1. Resource surveys, semi-detailed scale; including soil map.
	2. Land evaluation: (i) quantitative physical suitability for land utilization types; (ii) economic analysis.
	3. Selection of project; funding and decision to proceed.
Development survey of project	1. Resource surveys, detailed scale, or supplementary to those of feasibility survey.
	2. Land evaluation: suitability for defined land utilization types in economic terms.
	3. Preparation of project plan.

reflected in the soil classification, but also and more important, the sustainability of the farming practice cannot be measured in terms of an area to be planted to an annual crop. Decisions on appropriate land mapping units are not independent of the types of land use under consideration. The role of resource survey must be continued into the planning phase, so that information is provided on the experimental consequences of each proposed development. Moss (1968) coined the term "contemporary functional relationships". This requirement for a dynamic approach is in part a justification for computerisation, in that although very difficult to implement adequately in traditional survey procedures, the timeliness of output from a computer based information system, may enhance the chance of responding to this need.

Davidson (1965), criticises the ordering of events as given in the evaluation procedure. Davidson proposed a preliminary evaluation and economic analysis of land use options as an initial stage, to identify viable alternatives and their environmental requirements. These specific conditions could then be sought in the field, without wasting funding on a broad based resource survey with no clear intention in mind. In theory only useful information would be collected, although in practice taking this approach might mean future areas of good potential were ignored during a survey, because of temporary economic conditions. Young (1976) also introduces the approach of Beek and Bennema (1974), whereby a land utilisation type is regarded as a technical organisation unit in a specific socio-institutional setting to be as closely defined as the resources of the land. An iterative process of matching is then conducted to evaluate the land.

Desiderata for a land evaluation system have been suggested (Young, 1976; Brinkman and Smith, 1973) as given below:

"1/ The system should evaluate land for specified forms of use, defined as closely as the intensity of the study requires.

2/ The land use alternatives considered should be those which are not only physically possible but also economically and socially relevant.

3/ Evaluation should take into account both the production, or other benefits, from each land use alternative, and the inputs or costs necessary to achieve this production.

4/ The effects of the land use alternatives on the environment, particularly possible adverse effects (hazards), should be considered.

5/ The evaluation system should permit interpretation in stages, according to different purposes and intensities of survey.

6/ The system should be versatile, capable of adaptation to a wide variety of circumstances, both environmental and economic. For example, it should permit adaptation both to smallholder farming and to estate or other large scale forms of agriculture.

7/ The results of the evaluation should have a degree of permanence appropriate to its expected application; this means not only the period over which it will be consulted for planning purposes, but the anticipated duration of the planned changes in land use to which it refers. The results should not be unduly sensitive to short-term economic fluctuations.

8/ At the same time the system itself should be flexible, permitting periodic revision, for example with changing technology or with substantial and reasonably permanent changes in economic conditions.

9/ The final results of the evaluation are going to be read by economists, planners and those responsible for administering foreign aid. They should therefore be presented in terms which are simple, capable of being understood by the non-specialist, whatever may have been the complexity of the processes which led up to them. The presentation should be in terms which inspire confidence in government agencies and investment institutions."

The FAO Framework for land evaluation (Brinkman and Smyth, 1973) was designed to meet eight of the above criteria. By giving priority to economic viability, the requirement for permanence of the results, could not in Young's opinion be met. Of course with a computer based system, the latest land use plan using current economic information could still be presented, simply by updating the prices and costs.

Land suitability evaluation following the FAO system (FAO, 1976 & 1983) is a process of matching land utilisation types, with the land resources of a given land mapping unit, as illustrated in figure 1.1. There are two important concepts standardised in the FAO system, that of the land unit, and the land utilisation type. The definitions are as follows (FAO, 1983):

Land unit (LU)

> "A land unit is an area of land possessing specified land qualities and land characteristics, which can be demarcated on a map."

This term is synonymous with the land mapping unit referred to above. The land quality (LQ) and characteristics are defined as follows:

> "A land quality is defined as a complex attribute of the land which acts in a distinct manner in it's influence on the suitability of land for a specific kind of use. An example is erosion resistance. A land characteristic is an attribute of land that can be measured or estimated. Examples are slope angle and soil texture."

There can be interaction between characteristics, which is why the land quality is used in a comparison of land units with land use. The land quality is described by the state of a number of land characteristics.

Land Utilisation Type (LUT)

> "A land utilisation type (LUT) is a kind of land use, described or defined in a degree of detail

> greater than that of a major kind of land use, such as annual crops or grassland. In the context of rainfed agriculture, a LUT typically refers to a crop within a specified technical and socio-economic setting."

Figure 1.1 Showing how the rating for qualities of a LUT are matched with the quality ratings of a particular LU, for a hypothetical example (after FAO, 1983).

Land Utilisation Type: Maize, Mechanised production on state farms (Mozambique).

Land Use Requirements				Land Unit X	
	Diagnostic	Factor		Factor	
Land Quality	Factor	Rating		Rating	LQ/DF
		>600mm	S1		
Moisture		400-600mm	S2	500mm	Total
Availability:	Total	300-400mm	S3		Rainfall
Total Moisture	Rainfall	<300mm	NS		
		>120cm	S1		
	Soil	50-120cm	S2	150cm	Soil
	Effective	30-50cm	S3		Effective
	Depth	<30cm	NS		Depth

The principal of comparing requirements for a specific land use with the land resources of a discrete area, has been applied equally well in some of the earlier evaluation systems still in use today, such as the USBR system (1953) for irrigation classification. The improvement is in the accurate definition of how a land unit should be delineated. This offers a distinct advantage over the reliance on such a unit emerging naturally from the resource survey. Accurate description and definition of land units demands justification for how they are set, thus reducing the likelihood of one resource factor dominating the evaluation process.

A precise definition of the land utilisation type identifies the factors important to measuring the potential for that land use. Important criteria are not overlooked and there is a better chance of applying quantitative estimates to the final results.

A land use planning project normally necessitates undertaking a land evaluation, but land use planning is a broader concept. The land use plan involves taking the results of the land evaluation through, to at least the stage of recommendations. This entails the integration of physical; economic; financial; and often environmental evaluations, together with political and legislative constraints. A land use planning project may even involve implementation and subsequent monitoring. Land use planning can be formalised into a set of generalised procedures to be followed, but there is a danger in too much emphasis on standardisation. In practice there is no standard situation. The land use planner should always adapt to the local situation. This requires an understanding of local practice, customs and aspirations. Such knowledge is time consuming to attain and cannot be achieved as an office exercise. The desiderata quoted earlier do refer to the need to consider social and economic factors, but there is a possibility that in the adoption of strict procedures for physical evaluation, the more subjective socio-economic factors are ignored. This can happen because the planner having achieved a definite physical result only pays lip service to social factors, or he may be unsure how to incorporate the two disparate evaluations together. The situation could easily be exacerbated by automation, because the handling of socio-economic data requires a methodology based on logical inference, rather than manipulation of numbers.

1.3 Information Technology and Land Evaluation

The preceding discussion in section 1.2 illustrates that land evaluation can be a complicated task. Even where attempts are made to quantify the task in practice it is a very subjective procedure. To minimise the subjectivity, presentation of the justification for decisions becomes very important. Also the supporting data for a final classification map usually runs to several lengthy volumes. Land evaluation is a very lengthy process and the end results are often already dated when they appear. It is not unusual to find that policy makers who sanctioned the evaluation process, have been forced by the exigencies of their office to make decisions in advance of receiving the evaluation results.

It is the author's contention that the evaluation procedure could be improved by the application of some of the recent advancements ir IT. The advantages a computer based system would offer are listed below.

1. Data collection can be improved with less information lost during collection and processing.

2. Processing can be speeded up by automation of classification. Thus ensuring contemporary results are provided.

3. Standardisation can be introduced.

4. Other on-line information can be accessed directly and utilised efficiently.

5. Redundancy can be eliminated by updating of dynamic records and presentation of results in a useful and inexpensive form.

Together these measures have the potential to radically improve current land use evaluation procedures. The means to address the current deficiencies in implementation of land evaluation procedures were thought to be available through a variety of IT techniques. The intention of this work is to establish whether the proposed techniques are indeed feasible and cost-effective. The latter point has been given significant weight because of the interest in this work from a commercial company and the general requirement of the 'Aid' community, that an investment in new techniques should be economically justified and not a burden on the country concerned. In addition it became clear from the case study in Thailand that economic justification was a major hurdle in introducing new techniques. The observations on opinions and prejudices of consultant domain experts and local officials have made a significant contribution to the discussion of results in chapter 6 and are significant in considering the commercial implementation of the technologies investigated.

At the inception of this work, several IT techniques were receiving publicity as solutions to many information processing problems. Principally these were Relational Databases, Expert Systems; Artificial Intelligence Languages; Remote sensing , Image Processing and GIS. On the periphery of interest integrated packages and spreadsheets for economic and financial analysis, were also in vogue. The only software regularly used by the co-operating company in land use planning exercises were spreadsheet programs, although some attempts had been made to introduce the use of relational databases to the company.

The hypothesis was, that because these IT tools have been proposed as solutions to problems, without recourse to programming expertise, the adoption of a methodology to meet new criteria would be easily accomplished. The intention was not to build a complete information system to satisfy this aim, but rather to illustrate by building some components, whether or not the approach is feasible

Data collection is a major constraint to providing a useful evaluation system, typically many assumptions are made with no attempt to test their validity. One of the most common faults is the transfer of experience from one country to another when for subtle reasons it may not be appropriate. It can also be difficult to justify establishing an automated land evaluation system if there is not seen to be enough data to warrant automation. Usually, even though quite comprehensive data sets may be available, they are not in a form readily integrated into a computer system. Often a completely new data collection exercise must be instigated to establish a database of information in the correct format. This is obviously wasteful of past efforts and demanding on current resources. IT techniques which may aid in utilising archive data are thus of obvious interest.

One of the problems with the subject of rural land evaluation, is that much of the important information is in the heads of the real experts, the farmers. Richards (1985) notes that in West Africa attempts to develop the food-crop production sector by 'technology transfer' linked to advocating changes in land use, have had disappointing results. Richards believes local agricultural skills are under-utilised and considers it essential to harness the innovativeness of the peasant sector towards meeting national development objectives. Sadly this does not seem to be occurring in the land use planning sector. In Thailand it was assumed by most government officers that scientists or existing texts would provide the information for planning crop production and farmers tended to be ignored. One of the reasons for this is probably that no explicit procedures are laid down in FAO guidelines for the gathering of local knowledge, but it is also a difficult task to implement. Certain IT technologies may help in establishing systematic knowledge gathering procedures. In particular the induction systems discussed in 3.2.6, which may aid in knowledge elicitation. Even quite simple computer-based methods may be an improvement on current practice.

Relational databases have been available and understood for a relatively long time now in the IT world. Yet although the technology has been adopted, applications research seems to have been limited mainly to mundane payroll systems and similar tasks. The lack of publications on applications, together with personal observation of the use of systems in the company sector, suggests they are under-utilised. Where successful implementation has been publicised it has usually been for expensive mainframe or mini-computer systems in the financial services sector. On the micro-computer level, relational databases have tended to be used only as an information store and no more. Meanwhile even more powerful databases are being developed with knowledge-based techniques (Addis, 1985). The intention in this study is to investigate whether a proprietary relational database software package, could be used effectively for a knowledge based application. The dBASE data base management system (Ashton-Tate, 1984) was used for the purpose of this evaluation.

Artificial intelligence (AI) languages and expert systems are items of current research interest in the computer sciences and related disciplines, so their application was of particular interest. It was thought such programs would provide a "friendly" interface to the computer for subject experts. In addition A.I. languages are designed for symbolic processing, rather than very specific data types like integers or real numbers, this makes them well suited to modelling knowledge-based problems.

Image processing and remote sensing have also gone through an earlier phase of popularity but the significant change in the mid 80's was the availability of comparatively cheap micro-computer-based systems. Digital Image Processing is now widely accepted as a means of enhancing remotely sensed images and interpreting them to provide a variety of resource information. Of particular interest to the land use planner is the information which can be extracted on current land use.

Current land use is very important to any land use evaluation procedure, because without baseline data on the current situation change cannot be measured. Monitoring and the evaluation of progress in any implementation plan, are now widely recognised as essential activities for the success of rural development programmes. Remote sensing is then generally recognised as an essential part of a comprehensive land information system. For this reason the effectiveness of digital image processing of satellite data for land use planning is assessed in the particular context of the case studies chosen. The focus of the investigation was not on image processing techniques, but rather the value to the land use planner of a standard classification and the integration of such techniques, with other components of a land information system (LIS). The feasibility of basing a land information system on an image processing system is investigated (see chapter 6).

The nature of the work of overseas agricultural consultancy companies dictates that for most jobs the only viable option for using a computer system on short contracts would be at the micro-computer level. Mainframe machines would only be relevant to implementation of large scale projects, usually of national systems with substantial funding available. The declared interest of the collaborating company was in micro-computer based systems, so this study has concentrated on solutions which could be implemented on such systems. Minimising the investment required in both software and hardware are important considerations.

CHAPTER II

COMPUTER AUTOMATION OF LAND EVALUATION

2.1 Introduction

The automation by computer of land evaluation procedures has been attempted before. McRae and Burnham (1981) have commented that:

> "the entire operation of assigning land at a point or in some area, to a suitability or capability group or rating, can be committed to a computer, if the procedure can be completely specified in an objective manner and suitable data supplied."

This statement may be true, but in reality the conditions of a completely objective specification and availability of data are seldom met. A specification may be objective and complete as in the classification of soils for land bearing capacity by observable physical parameters, (Hartnup and Jarvis, 1976), and for similar narrowly defined classifications, but land evaluation for agricultural use is a very subjective process, and land use planning in general tends to be even more subjective. Even with narrow terms of reference, a spatial classification will be limited by the suitability of data available. Because of the problem of relying on a few samples to represent an area, facts will be 'fuzzy', with an uncertainty attached to the data and hence the result, even if the model is defined in detail. Often the model itself is incompletely understood.

2.2 Existing Systems

Rudeforth(1975) developed a computer program to assign sites to categories of the land use capability classification and to determine the suitability of sites for growing barley and early potatoes for Pembrokeshire, Wales. The intention was to construct a land classification map at a 1:50,000 scale. Compared with typical land evaluation tasks in developing countries, a very small area of about 1000 Km^2 was covered . The land capability classification used was that of Bibby and Mackney (1969) based on site, soil and climatic factors. Physical factors and their limits used in the classification were taken as guidelines and not definite criteria. One of the purposes of applying the classification and comparing the results with current land use, was seen by Rudeforth and Bradley (1972), as that of identifying discrepancies between actual land use and supposed potential use, to highlight inappropriate limits, or indicate other relevant factors, so as to progressively improve such classifications. Even with a detailed survey of almost 1000 soil pits at 1 Km intervals, and considerable support data in the form of geological maps, topographic maps and air photographs, the results were very subjective and the complete neglect of socio-economic factors was an acknowledged deficiency. For example the high value early potatoes crop was actually grown on inferior land, in order to catch the higher prices of an early market. This illustrates the point that, even in the 'West', in land evaluation it is still likely to be some time before the procedures are completely specified in an objective manner, so that computerisation of the processes is not a trivial matter of commissioning a program to be written for the task. Consideration must be taken of how the classification is likely to be developed and how it will be improved.

Programming of land evaluation procedures must be achieved in a flexible manner which can be easily amended and improved. Data should be independent of the procedural programming so that a standard approach can cover a wide area, with differing key criteria. Yet to accommodate the change in the factors of importance over a large area, automation must allow for the identification of the crucial criteria. This is by no means a simple specification to meet, particularly with traditional programming languages used so far in attempts at automation. One possibility is for a declarative rather than procedural programming style as discussed in more detail in section 3.2. Declarative style is based on a set of logical axioms making up a program. Computation is then a constructive proof of some stated goal, following the rules of deduction, rather than following a fixed pathway. This differs from a procedural style whereby every relationship must be strictly defined and implemented in a method closely aligned to the underlying processing capabilities of the computer, rather than to logic. So called fourth generation languages (Bate and Vadhia, 1987), including relational databases and some A.I. languages, might meet this requirement(see Chapter III). But so far automation has mainly been implemented in the procedural languages of FORTRAN, PASCAL and BASIC.

The FAO system of land evaluation was computerised by Wood and Dent (1983) with a methodology developed called LECS, "A Land Evaluation Computer System", implemented on a mini computer in FORTRAN. A modular approach was adopted and an objective was to allow users a maximum of control over the selection of agronomic and economic control data. FAO projects (Kassam et al., 1982) have used an extension of this methodology but it has not been widely adopted. One of the reasons for this, is probably the difficulty in understanding and maintaining lengthy FORTRAN programs. In the author's experience it is very difficult to adapt another's code for one's own use. Most programmers prefer to write their own code. The problems in altering and debugging a FORTRAN program would make distribution to external users (to the FAO) a problem. In practice a system always seems to require modification for a new situation. LECS was intended to be criticised and developed through testing and implementation by other specialists. This has happened to a limited extent, as with the work of the International Institute for Land Reclamation and Improvement (ILRI) with an Indo-Dutch research project called ILWIS (Integrated Land and Watershed Management Information System) for which one of the objectives was an application of LECS. Even though LECS had been established for some years, it was not possible to implement management and socio-economic considerations in assessing crop suitability, because that module was still not available on IBM-AT compatible computers. The LECS Pc version had to be modified to use data directly from a relational database and various problems were encountered running LECS. These seem to have been caused by inappropriate models for the new situation. Predicted yields from LECS were modified by coefficients obtained from validation with actual yield data and linear regression analysis. This would indicate the model was not an accurate representation of the situation and perhaps what is really needed, is a fast means of modelling a land use system, rather than a fixed framework which is constantly adapted. The adaptation approach will always require highly skilled programmers and subject experts and will consequently be an expensive task. If appropriate computer tools can be identified for use by subject experts, programming and software maintenance can be minimised. Eventually LECS may be established as a fully portable comprehensive land evaluation system, but because of the choice of method and enormity of the task this will continue to be a very slow process.

The soil survey of England and Wales (now the Soil Survey and Land Research Centre) have established a land use information system for the U.K. (England and Wales) called LandIS. This was developed to it's current state over a period of about 8 years, with a team of 4 people to coordinate the tasks. Soil information can be accessed on a 10 Km grid square basis (Pers. Comm.). Climatic data on a 5 Km grid square basis is also integrated in the system as well as topographic information. The 1:250,000 series soil maps for England and Wales have been digitised and are accessible through the system. The system is based on a VAX 11/750 with 670 Mbytes of disk storage. A database supplied by DEC called DATATREIVE was used with other software including a common DATA DICTIONARY. Routines have also been written in PASCAL. The system is not powerful enough to support the remote regional structure of the soil survey, but it is currently being upgraded from the VAX 700 series to the 800 series. The database will also be changed to a powerful relational database. This will allow compatibility with a range of computers even down to micro-computers. The interface with the system is reasonably 'user friendly' though not very flexible, depending on prepared question screens, which are linked in a fixed hierarchical order. It is not possible to jump to another part of the system directly from a deeply nested query, the user must work back through previous option screens, unlike some other systems for instance ERDAS. Nor can a query be made directly by the user, although this will presumably be ameliorated to some extent by the implementation of a database such as ORACLE, which supports Structured Query Language with the relational database, though even then use of such a language requires expertise. The Soil Survey is also considering the contribution that could be made through use of AI languages or techniques. The main deficiency recognised by the Survey at present is the poor interface to available graphics software but this is being remedied. It is the Survey's opinion that an experienced team of about five specialists could implement such a system in about two years with a minimum of data included. Addition of new data is then done progressively as it is collected. Processing historical data into the system can be difficult to achieve because of validation problems, as old data was collected to a more subjective and inconsistent format.

The Soil Survey's experience shows that implementation of National systems is not a trivial task which can be accomplished quickly. Every situation is different and thinking time will always be required to arrive at a well structured and flexible system. It would seem the Survey is now faced with a complete change of software and hardware, which will undoubtedly be an expensive exercise. This probably could not have been avoided because of the rapid change in computing over the development period of the system, but the importance of identifying flexible and easily maintained software is evidenced by their experience (pers. comm.). Significantly the Survey are now also using micro-based systems, in particular the TYDAC mapping system, for some of their project orientated work and intend to interface this with their LandIS.

One of the problems of automation the Soil Survey experienced was that of collecting data in a suitable form. Having established a reliable evaluation system the Survey is still lacking the detail it would prefer, even after a major data collection exercise for the production of the 1:250,000 series maps. That data was collected according to a well defined computer compatible coding system over two to four years and was then keyed in and validated. The data was collected for every 10 km grid square but augerings were at random, so for some squares the data is not uniformly distributed, but is more usually clustered. With the considerable variability of soil type that can occur over a short distance this leads to obvious difficulties in giving practical advice to farmers. The other problem, as yet unsolved, is how to deal with historical records which are not yet in the system, validation of such data can be extremely difficult because the descriptions of soils were not always made to a uniform standard in the past and records are often incomplete, but at present there is a large source of potentially valuable data left inaccessible to the computer.

The computer language used in the Soil Survey System, PASCAL, although designed as a teaching language, does embody many of the concepts of the most powerful high level languages. It is a "structured" language whereby the problem is broken down into it's constituent parts and code is easily understood. Thus the choice of this language by the soil survey was not unusual. Surprisingly some information systems are still being written in rather outdated and restrictive languages such as BASIC. Improved versions of BASIC are now available for micro-computers (Shammas, 1987) but traditional BASIC, first developed 25 years ago, does not support structured programming, coding can be contorted to fit the limited commands available but then the program is not easily understood and maintained by others. Most BASIC versions lack callable subroutines, multi-line functions and local variables : until some of the latest versions, it was also an interpreted language and consequently very slow for handling large amounts of data. Despite these deficiencies it was chosen as the language for implementation of the Fertility Capability soil Classification system (FCC) as developed by Buol (1972).

The FCC was developed in an attempt to bridge the gap between the sub-disciplines of soil classification and soil fertility. The FCC users natural soil classification systems such as soil Taxonomy (Soil Survey Staff, 1975) and groups soils in a quantitative manner according to their fertility constraints. The system is supposed to be directly applicable to FAO's (1976) Land Evaluation Guidelines (Sanchez, 1982). Although the FCC is concerned exclusively with soil factors it can be incorporated into the FAO's framework for Land Evaluation (1976). The FCC system consists of three categorical levels: type (topsoil texture); substrata type (subsoil texture); and 15 modifiers covering such factors as gleying; moisture regime; acidity; salinity and presence of essential elements. The soils are classified by determining whether the characteristic is present or not. Each FCC type, substrata type and modifier fits FAO's definition of a land characteristic (see glossary). The use of the FCC is claimed to eliminate the problem of interaction between characteristics. Of the seventeen land qualities related to productivity or plant growth listed in the FAO framework, seven are related to FCC parameters: nutrient availability; salinity and alkalinity; soil toxicities; oxygen availabılity; adequacy of footholds for roots; and resistance to soil erosion (Sanchez et al., 1982).

As an attempt a provide a universal standard for a quantitative soil fertility classification, the FCC is commendable, but the method of automating the classification has been less well planned. The system is implemented and distributed to other researchers in BASIC. One of the apparent ideals of the FCC, that of a universal system, leads to problems in the choice of BASIC. Altering a system written in BASIC is not easily achieved since the limits and variables used are an integral part of the procedural programmes. Considerable debugging is likely to be required even after information has been altered or added.

The concept of a universal system to be applied globally, is probably flawed anyway, in that conditions vary so widely from country to country, that what is really needed is a system by which limits can be easily altered and the system 'trained' to new conditions. This has been recognised by the FAO in emphasising the concept of a framework for land evaluation rather than fixing the factors and limits to be used. In Thailand, some of the limits used in the FCC system were found to be inappropriate, indeed some factors had not been included at all (pers. comm.). What is required is a system where the data is independent of the system model, as is attempted with the 'inference engine' of an expert system (see section 3.2), or the structure of a relational database. If the FCC is to be integrated as a part of an FAO based land evaluation system the use of BASIC would also complicate the interface required, as with an interpreted language like BASIC it would not be possible to call the fertility program directly as a sub routine.

An alternative would be to use an expert system approach for modules to the main land evaluation task, such as soil fertility assessment. This has been done for estimation of lime requirements in the humid tropics (Tropsoils Project, 1986). The Tropsoils project, whose goal is to develop improved soil management technology for developing countries in the tropics, developed an expert system called ACID3B for assessing lime requirement using EXSYS. EXSYS is an expert system shell of the rule-based type (see section 3.2.2.4). Because the knowledge base in EXSYS does have an affect on procedure, modifications do have to be made with care, but programming effort is minimal and the rules are stated almost in their natural English form. As with most of the systems discussed fairly standard query screens are displayed, but a complete natural language interface is not provided. An advantage over the BASIC written FCC system is that a 'why' facility is available; now widely regarded as essential to a good expert system. The 'why' facility provides justification for an answer to the user on request, by providing a display of the rules triggered by the answers, given at a particular stage in the query. EXSYS systems such as ACID3B are easier to modify and develop than something like the FCC using BASIC, but unfortunately as with most expert system shells integration as part of a large system written in other languages, it is even more difficult to achieve than with BASIC.

The ILWIS project (Valenzauela., 1986) has to some extent improved on previous attempts at automation of land evaluation and related tasks. It can actually incorporate the LECS system as it is modular based. ILWIS is a computerised system to assist comprehensive land and water resources planning. The key component in linking modules is the relational database called ORACLE (RSI, 1980). It is implemented on an IBM-AT compatible micro-computer, with an 8-bit colour graphics card to provide limited image processing of remotely sensed data and handling of digitized maps. Only a few of the models envisaged have been completed but the results from the land model showed the capability of the system to generate useful information for planning. Satellite data was integrated in the erosion model although it was found impossible to obtain an adequate land use map by automatic classification. This was because the vegetation and land use of the regime concerned, consisted of complexes with a spatial resolution often an order of magnitude less than the spatial resolution of the satellite data.

Spatial representation of data in a computer automated land evaluation system, may be handled by either raster or vector methods, or a combination of the two and the pros and cons of these method, are considered in more detail in section 3.6 on GIS. The raster based methods have found favour with most of the attempts at automation so far implemented, as in those for pedological assessments (Webster et al., 1979; Moore et al., 1981; Rogoff, 1982); Bie and Schelling (1977). Millington (1986) cites the following as advantages of the grid square (or raster) approach :

1. Grid square matrices can be generated by relatively trivial programmes.

2. Grid meshes are hierarchical and, unlike irregular polygons, can be easily aggregated.

3. Grid meshes are superimposed on commercial maps and therefore provide a readily available grid framework.

4. Considerable advantages accrue in computerized data storage and retrieval if data relates to a grid structure.

5. Overlay comparisons and statistical testing are undertaken easily.

6. Vector data on a grid format is easily linked with raster based satellite pixel data in more advanced information systems.

Millington (1986) also notes that the combination of physiographic, socio-economic and remotely-sensed data need to be urgently investigated using geographic information systems.

2.3 Summary

Computer automation of land evaluation has to some extent already been attempted. The problems identified from previous attempts are: a dependence on expensive computer hardware; an inflexible approach; a requirement for programming expertise; problems of integrating modules; and a need for a means of combining spatial data sets. These problems of technical implementation have to be considered in the context of a typically incomplete problem specification, because the model is seldom wholly understood and because of incomplete or totally absent data sets.

CHAPTER III

INFORMATION TECHNOLOGY TOOLS

3.1 Introduction

This chapter gives the background and current status of selected IT techniques used in this study. The techniques discussed are Databases; knowledge Based Systems; Logic Programming; Image Processing; and GIS. This review of the techniques concentrates on the features of particular importance to the land evaluation application.

3.2 Databases

Information stored on a computer in an accessible way is termed a database. The software to use or modify the database is termed a database management system (DBMS).

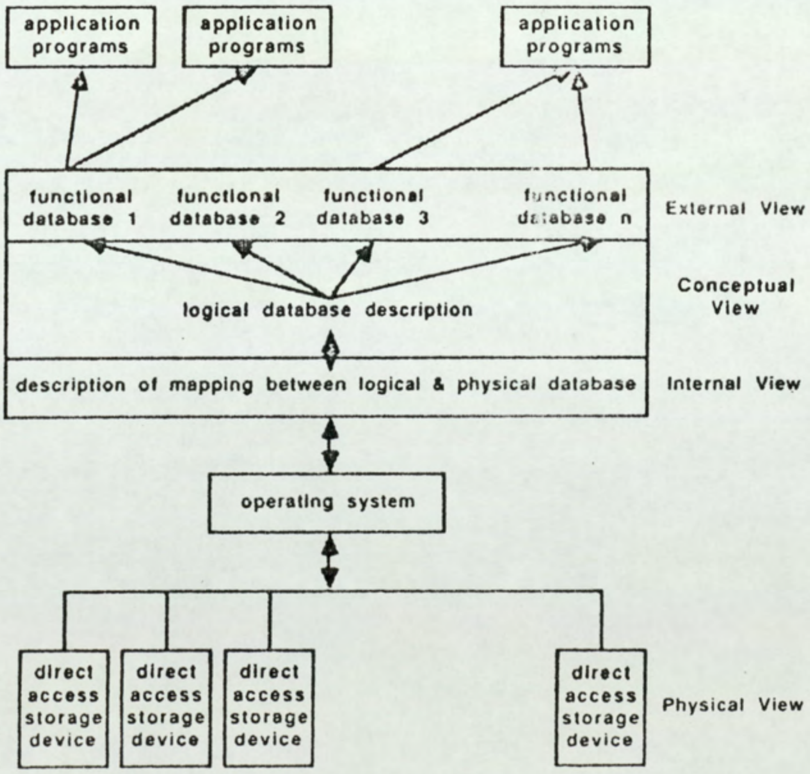


Figure 3.1 The Standard DBMS model (after Bate & Vadhia, 1987).

The sophistication of DBMS's varies, but one of the major objectives of such systems is to allow users to query and manipulate the system in a straight forward command language, without the necessity to understand the underlying data representation, or the algorithms required to maintain the integrity and validity of the data.

A working group comprising the American National Standards Institute (ANSI), the Committee on Computer Information Processing (X3), together with the Standards Planning and Requirements Committee (SPARC), reported in 1978 on a basic framework for all future DBMS. This proposed an architecture independent of any particular data model, as shown in fig 3.1. The four levels (schema or views) of planning or abstraction in a system, are shown in the figure: external schema; conceptual schema; internal schema and physical schema. The external schema is the view of the database as seen by the user. The external schema is also known by the term functional database. The conceptual schema defines the relationship between logical records and storage files. The physical schema is specific to the particular hardware and software used.

In formulating a query to a database system a query language is used, which is often of a standard form, supported by several commercial databases, as with the Structured Query Language (SQL) developed by IBM. To achieve this, the query is handled by a query processor which interprets the query language into commands which will be understood by other parts of the DBMS, utilising the database description, written in a data definition language (DDL). The database manager module then translates the query into terms that the file manager module can understand i.e. operation on files rather than abstract structures.

In a DBMS, between the level of the computer dealing with bit manipulations and the user formulating a query on the particular subject matter, there can be many levels of abstraction (Ullman, 1982). Only the physical database exists, as files and records on a permanent storage device such as a hard disk. A conceptual schema is an abstraction including only facts and structural rules which describe the environment for which records are stored. The conceptual schema describes the static knowledge and indicates clearly how elements relate.

Major advantages of a DBMS are (Bate and Vadhia, 1987):

1/ Data Independence between the program (logical view of the data) and the data (physical database).

2/ Minimising of data redundancy by avoiding duplication.

3/ Data security by access control.

4/ Concurrence control to allow simultaneous use.

5/ Easy data access reducing requirement for coding of applications by programmers.

6/ More efficient storage of information than in a file system.

7/ Maintenance of the database is separated from application programs.

3.2.1 Relational DBMS

Commercial database systems have used one of three main data models in their design: relational; network or hierarchical. Early commercial database systems favoured the network or hierarchical models, but Ullman (1982) expected the relational model to prevail and this has indeed occurred. Ullman identified two main reasons: First that the ease of use of the relational model, is superior to the other models and secondly that problems in efficient implementation of the relational model, could be solved by physical implementation ideas from the other two models.

The mathematical concept underlying the relational model is the set-theoretic relation, which is a subset of the cartesian product of a list of domains (Ullman, 1982), where a domain is simply a set of values. The cartesian product is the set of all combinations of values from the domains considered. The domains can be given a name and are also referred to as attributes. The members of a relation are called tuples. Thus in the diagram in figure 3.2 we have a table representing the relation for some soil records (extracted from a dBASE III file holding data on soils in Thailand). The attributes (or domains) are 'SOILNAME', 'SYMBOL', 'USDA,' 'SLOPE' and 'DEPTH', and each row or record can be termed a tuple, which can be viewed as a mapping of attribute names to values in the domains of the attributes.

Soilname	Symbol	USDA Class	SLOPE	DEPTH
CHALONG,GRAVEL	Chl-gr	OXIC PLINTHUDULTS	4	125
NONG KLA	Nok	TYPIC PALEUDULTS	5	96
PHATTYA	Py	QUARTZIPSAMMENTS	3	150
BAN THON	Bh	TYPIC TROPOHUMODS	4	185
RAYONG	Ry	QUARTZIPSAMMENTS	2	120

Fig. 3.2 Showing a relation from a soils database.

SOIL (SOILNAME, SYMBOL, USDA,) is an example of a ternary relationship from the above table, for which SOILNAME might be what is termed a key attribute. SOILNAME would then appear in other entity sets or files. Operations may be performed on the relations such as 'SELECTION' of tuples that meet a certain criteria or a 'JOIN' of relations. The relational model involves both relational algebra and relational calculus. Relational algebra is a procedural language used to specify the operations to be performed. Relational calculus is a non-procedural query language in which the properties of the information to be operated on can be specified, rather than the detailed steps by which the operation is to be achieved (Bate & Vadhia, 1987). The manipulation of relations is through relational algebraic operators such as : Union; Cartesian Product; Projection; and Selection (Codd, 1970). A Union or Join builds another relation from two given relations, such that the resulting relation contains concatenated pairs of records, that satisfy a given condition. Projection produces a specific attribute from a given relation. Selection can be considered as choosing one or more rows from a relation.

Some key points concerning relational databases are (Bate and Vadhia, 1987):

1/ The language used to access data is non-procedural i.e. describes what data is wanted rather than how to get it.

2/ The language is set-orientated rather than data-orientated.

3/ Efficient search algorithms are essential to the relational model.

4/ Key data is duplicated.

5/ The database structure can be changed in a way that is transparent to the user.

The separation of application programs and data in a relational database; ease of data access; set orientated language; and concurrence are all features of potential benefit to the automation of land evaluation procedures.

Automation of land evaluation to establish a complete land use planning system is a complicated and lengthy procedure, as the experience described in section 2.2 shows. Equally a vast amount of data is generally called upon in the active operation of such a system. Digitizing the required data is normally a bottleneck, whether it be archive data or newly acquired information. If applications programs are not independent of data structures and physical storage, the input of data has to be delayed until a supposedly complete system is designed. Inevitably modifications will still arise, which may be complicated by data dependence. In a relational database there is only one data type, the relation (Ullman, 1982), and logical data independence ensures that, unless information required by an applications program is deleted, the conceptual database can be modified to accommodate new information without rewriting of applications. So applications development and assembly of a database, can proceed in parallel and modifications to either process, will not have ramifications on the other parts of the system.

Easy data access is important because this also speeds up the process of assembling a database. Teams of data processors can be used, without great concern that files will be lost and validation of data entry is simplified.

A set-orientated language simplifies data independence, because in contrast to a data-orientated language, where a looping structure (e.g. WHILE DO) would have to be coded, with a count or some conditional check, a SELECTION would not be affected by the number of records.

Concurrence means that many users can query a DBMS at once and actually alter and update the data for a national land use planning system: this would certainly be important if it was to be maintained up to date.

The advantage of physical data independence raises issues of the portability of a DBMS and data between different hardware systems and this is certainly an important issue, but for this work interest has centered on micro-computer systems and specifically IBM-compatibles. The supposed advantages of a relational DBMS, were of significance in automation of land evaluation procedures. Concurrency was not an issue examined because a networked system was not available.

3.3 Knowledge Based Systems

Knowledge can be defined as assured belief (Kirkpatrick, 1983). Epistemology has traditionally involved a search for certainty (Hamlyn, 1970) and indeed skeptics have questioned whether knowledge is possible at all, but the theory of knowledge will not be argued here. The knowledge referred to in computer automated Knowledge Based Systems, or Intelligent Knowledge Based Systems (IKBS), is normally domain specific and of a specialist nature. Simons (1984) proposes that "Knowledge is information that the computer can think about". Addis (1985) points out that from the information technologists point of view, concern with the criteria of certainty is irrelevant. The technologist must ensure his machine-system does not distort the 'knowledge' but manipulates and presents the 'knowledge' in a consistent manner. Addis considers that 'knowledge' is what the user designates it to be; the technologist need have no opinion.

Intelligent Knowledge Based Systems follow a design approach which has emerged from research into Artificial Intelligence. The principal underlying the new design objectives of IKBS is that a better problem representation can be achieved by engineering 'knowledge' rather than 'data' (Stefik and Conway, 1982). Choice of knowledge representation is critical to IKBS, because the way representational features can be manipulated and queried effectively defines the scope for perception, understanding and knowledge in such systems (Simons, 1985). Simons identifies the tasks involved in knowledge representation as:

1) Structuring the explicit knowledge in a suitable form.

2) Encoding the rules for inference.

3) Specification and control of semantics and inference.

4) Accommodation of new and incomplete knowledge through flexible design.

5) Knowledge elicitation.

Within the A.I. research community it is seen as the task of the knowledge engineer to implement the above tasks. Feigenbaum and McCorduck (1984) give guidelines for the work of a knowledge engineer:

- You can't be your own expert.
- You must be prepared to discard initial efforts.
- The problem must be appropriate for the techniques.
- You need to meet the expert more than halfway.
- Use tools that work.
- Weighting procedures are needed to cope with uncertainty.
- AI systems need to be able to accommodate new knowledge and delete out-dated information.
- The problem needs to be interesting (i.e. not trivial).

Addis (1985) suggests knowledge representation does not depend upon any particular mechanism, nor rely upon a unique theory. The problem is not the generation of specialist programs, but the induction of new theories.
A particular type of knowledge based system is the expert system, with a rather closer definition than many systems said to exhibit knowledge. A large potential market in expert systems is perceived , with a forecast of a 350 million dollar market in such systems by 1990 (Harmon et al., 1988).

3.3.1 Knowledge Acquisition

In developing an IKBS, knowledge on a particular domain of interest has to be extracted and represented to develop a conceptual model. Until the conceptual model has been derived a reliable system cannot by designed (Hart, 1986). Knowledge acquisition may be by observation from existing records, by questionnaires or by interview, as in systems analysis. Knowledge elicitation, whereby the object is to 'capture' the knowledge of a subject expert, has been identified as the major bottleneck in developing expert systems (Feigenbaum and McCorduck, 1984; Welbank, 1983). Specialist methods for knowledge elicitation have been developed (Hart, 1986).

Even in a standard systems analysis approach, adopted in the modelling of any complex system by computer, it is now recognised that the subject expert, or ultimate users of a system, should be involved at the early stage of building a conceptual model. This strategy is usually essential to an expert system approach where the system is not likely to be well understood. As already mentioned in section 3.2 it is widely believed an expert should not be his own knowledge engineer. Hart (1986) gives two reasons for this:

- He will usually have insufficient knowledge about programming and expert system techniques

and

- He will find it difficult to describe his knowledge completely and correctly.

Yet it is contended that the programmer could act as the knowledge engineer provided he has the right interpersonal skills (Feigenbaum & McCorduck, 1984; Welbank, 1983). Such skills are listed by Hart (1986) as: Good communication skills; Intelligence; Tact and diplomacy; Empathy and patience; Persistence; Logicality; Versatility and inventiveness; Self confidence; Domain knowledge; and Programming knowledge. Hart acknowledges it is unlikely that a single knowledge engineer would have all of these qualities. This perspective seems to envisage a roving knowledge engineer, able to apply himself to almost any particular problem, much as has been expected of the systems analyst. It would still seem to be a matter for contention whether this is actually a step forward. The only skill listed by Hart which subject expert may be presumed not to have, iꟳ that of programing expertise. This too is changing as more technical experts accumulate programing experience.

The elicitation stage should provide a complete and correct description of the expert's knowledge and the way in which that knowledge is handled.

3.3.2 Expert Systems

Harmon et al. (1988) believe that in the long run, expert systems will allow managers and subject specialists who are not programmers, to develop and run computer programs, without having to understand the underlying program. It has been suggested by Harmon et al. (1988) that this will be the most profound change to result from the current batch of AI techniques and procedures.

A formal definition of expert systems as approved by the British Computer Society's committee of the specialist group on expert systems (Naylor, 1983) is :

> "An expert system is regarded as the embodiment within a computer of a knowledge-based component from an expert skill in such a form that the system can offer INTELLIGENT ADVICE or take an INTELLIGENT DECISION about a processing function. A desirable additional characteristic, which many would consider fundamental, is the capability of the system, on demand, to JUSTIFY IT'S OWN LINE OF REASONING in a manner directly intelligible to

> the enquirer. The style adopted to attain these characteristics is RULE-BASED PROGRAMMING."

Expert systems adopt a declarative program style, in other words, if all the relevant aspects of a problem are declared by the user, the program will then find a solution without being instructed as to how to use the information, as in conventional "procedural" style. The logic embodied in the knowledge base of rules, constructed by the subject expert, determines the path taken by the program. Some computer languages such as Prolog have an in-built declarative control structure, which is why some people would suggest such a language should be used in expert systems. However this is not an essential requirement, declarative expert systems can be written in conventional procedural languages, such as FORTRAN or BASIC (Naylor, 1983).

The efficiency which expert systems exhibit in avoiding repetition of questions and shortening the search path, by asking only pertinent questions is one of their major advantages. If a system is not based on logic with a declarative control structure there is a danger that questions will be repeated, even if some kind of hierarchical structure has been achieved in a procedural language, by judicious use of "IF THEN" type statements. It can also be very difficult to identify and code all the explicit links required in a large complex system.

The other facilities offered by expert systems such as explanation and certainty measures, would both be useful additions to the classification procedure. For example, users of the FAO evaluation guidelines (as interpreted for the Thailand situation), are often perplexed as to how exactly a particular site should be rated for certain factors. In assigning something like soil texture categories, there may be some doubt as to which standard texture classes should be grouped together. In such a situation, advice as to how a question is to be interpreted is extremely useful. This would be available at any point in the system and additionally, explanation can be provided as to why a decision has been made. The latter shows which rules have been used to arrive at a particular outcome. Showing how the information is to be used often clarifies how the question should be answered. It also satisfies the user as to whether or not the correct logic has been applied.

Certainty measures express the degree of confidence the user can have in the final decision, reflecting either uncertainty as to facts or the logic of the system being considered. In the case of land evaluation uncertainty as to the logic derives from doubt about the influence of some factors. Uncertainty as to facts reflects the level of confidence in data collected on a particular land unit. Often sampling of land units, to collect data on attributes for evaluation, is not as comprehensive as would be desired. Certainty factors for each attribute would indicate the level of precision for the data used. Such a facility could be very useful in identifying regions where either the system is incompletely understood, or the data available is not yet sufficient for very reliable classification. In Thailand at present, information on detailed crop requirements is often based on work in other countries, or on a very few research results. The strength of belief to be placed in these requirements could be expressed using certainty factors.

Expert systems to accomplish a particular task can be programmed directly in a chosen language, but they are more usually met in the guise of expert system shells. These shells can be likened to an original expert system designed for a special purpose, with the information specific to that purpose subsequently removed. The structure and mechanisms can then in theory, be applied to any appropriate task someone else has in mind. Instead of having to program directly in a language, the non-programmer can then use such a shell to logically design his system. Some shells can handle uncertainty others cannot, and the provision of explanation is also variable. Most systems require the user to devote some time to learning how to use them but working expert systems can be built very swiftly by subject (domain) experts themselves without recourse to computer specialists.

Most expert systems can be divided into two distinct parts, i) the knowledge representation and ii) the control structure. Systems differ in the techniques used to implement these parts and this is discussed briefly in the following sections.

3.3.2.1 Expert System Structure

Any expert system is characterized by three fundamental elements according to Simons (1984): The Knowledge manager; the Knowledge Base; and the Situation Model.

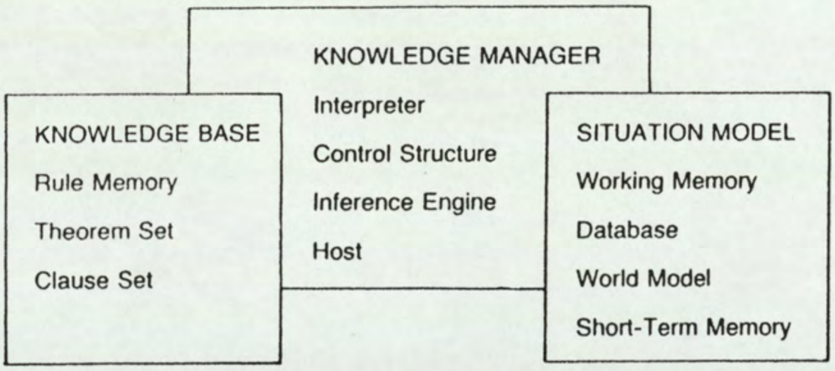


Figure 3.3 The Main Elements in Expert Systems

Fig 3.3 illustrates the model, with alternative names listed for the components (d'Agapeyeff, 1983). The separation in this model between the knowledge base and the situation model is significant, some system descriptions merge the two parts, but the distinction is really quite important.

The knowledge base contains rules and facts concerning the particular application domain (in this case land evaluation). The facts in the knowledge base will be constant for the specified conditions. For instance average daily temperature might be constant over a province, this could be included as a fact. Rules will cover important instances of particular variable values.

The situation model will store the current instances of variable values to be considered through the knowledge manager in relation to the knowledge base. The knowledge manager essentially determines the order in which rules are tested, to interpret the current contextual data in the situation model.

Knowledge base seems a fairly universal term at this date, but control structure and Database seem to now be the preferred terms for the knowledge manager and situation model respectively, so they will be used here.

3.3.2.2 Semantic Networks

Separation of the knowledge base and control structure is not always distinct, as in the case of the semantic network used in some systems for knowledge representation - such as PROSPECTOR (Duda et al., 1978) A semantic network is based on the idea of associations between concepts, as in the notion of 'associative memory' suggested by Aristotle (Ackrill, 1981; Alty & Coombs, 1984). The basic unit in a semantic network is a structure of two nodes linked by an arc. Each node represents a concept and the arc represents a relation between the pair, for instance:

```
                    has
Land Unit 2 -------------------> Soil type - Ban Bung
```

The arc direction preserves the subject/object relation between the concepts. The appeal of the semantic network seems to be the simplicity with which correct deductions can be made, once a network has been generated (Simons, 1985).

3.2.2.3 Frames

A frame design is essentially a semantic network in which nodes are represented by frames instead of specific objects (Minsky, 1975). A frame is a complex data structure which describes several attributes of a particular object or situation. As for example in Fig. 3.4.

```
 ________________________________________
|  Land  Unit : 2                         |
|        Soil Type  : Ban Bung            |
|        Topography : Undulating          |
|        Drainage   : Poor                |
|        Cropping   : CROPS               |
|_________________________________________|
```

Figure 3.4 Illustrating a hypothetical frame of a Land Evaluation System

The frame in the figure is instantiated to a particular instance i.e. Land Unit 2, this would be a unique identifier used to access other frames, for instance a frame called CROPS, to give information on suitable cropping systems for that land unit.

The top levels of a frame are fixed and represent things that are always true about the supposed situation. Lower levels have many terminals (slots), which must be filled by specific instances or data. Each terminal may specify conditions it's assignments must meet. Different frames share the same terminals and this is critical to coordination of information gathered from different viewpoints. Frames are used to represent a dynamic situation, so the contents of slots in a frame will change over time.

Minsky argues that traditional logic cannot deal very well with realistic, complicated problem, because it is poorly suited to represent approximations to solutions and these are absolutely vital. Minsky contends that thinking always begins with suggestive but imperfect plans and images, these are then progressively replaced by better, but usually still imperfect ideas.

3.3.2.4 Production Rules

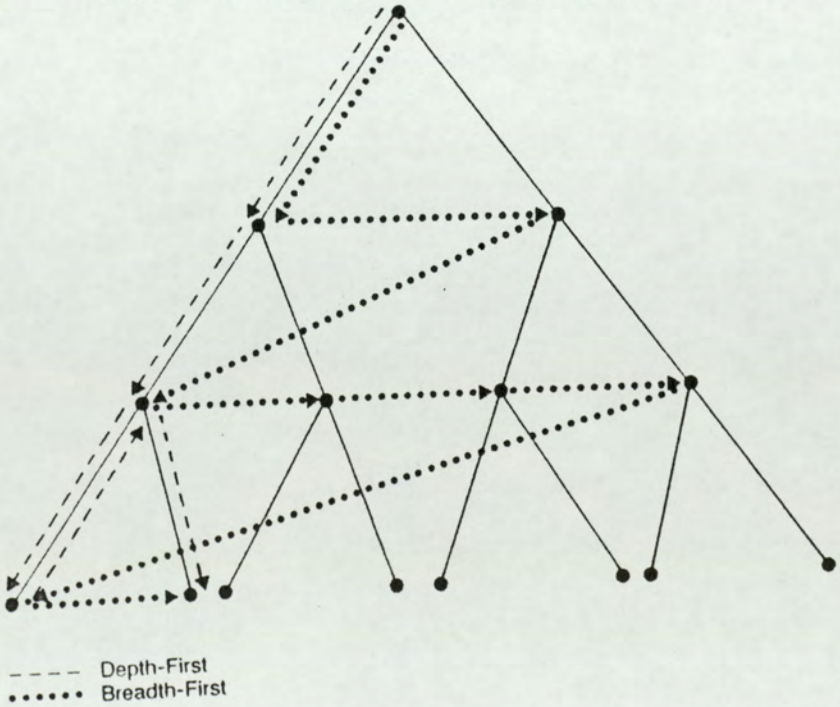
Production rule systems are the most common form of commercial expert system. A production rule is simply a structure of the type "IF (condition) THEN (action)". A collection of such rules together with established facts can be used to build up a knowledge base. The production rules define a set of recognised relationships. The database representing the current problem is then matched against one side of the rules. Rules can trigger other rules where the "(action)" of one rule is the "(condition)" of another, but a control structure is also needed to resolve choices between rules with the same conditions, or for direction to other parts of the knowledge base. The Micro Expert system used in this study is an example of one type of rule based system (see section 4.2).

3.3.2.5 Control Procedures

Whatever the nature of the knowledge base adopted, some kind of control structure will be required. In the discipline of Artificial Intelligence, control structure is often described in terms of a 'state-space search' (Alty & Coombs, 1984). The theory is that it is possible to describe all possible sequences in a problem solving situation and this can be represented as a tree. A solution to the problem results from the application of rules, triggered at a given state, to produce a transition to the next state. In a complex problem it is necessary to have a means of deciding which rule to apply to a state, in order to progress towards the solution or goal. The general strategy governing such decisions is known as the control structure. The order in which rules are checked (matched against the database) is usually important. All possibilities could be considered by checking every rule or node, but this would not be very efficient and may result in problems, such as the repetition of questions to a user.

Search direction can affect the efficiency of a search: two terms often used are backward-chaining and forward-chaining. Backward-chaining (top-down), involves moving from goals to start state and forward-chaining (bottom-up), moves from start state to goals. A state space may also be examined either depth-first, or breadth-first as shown in fig 3.5. A depth-first search pursues a particular path, until a specific route through the tree is resolved, or failure occurs at a node or rule, and then the next path is explored in depth. A breadth-first search generates all possible alternatives at a given level and then all alternatives at the next level and so on.

Figure 3.5 Depth and Breadth-First Search.



Whichever search strategy is adopted there is a flaw in the state-space theory, in that for many and perhaps the majority of practical situations the complete state-space is not known, either because the problem is incompletely understood or because of a combinatorial explosion in the number of possible outcomes. For example Haugeland (1981) has estimated that for the average chess game there are likely to be 10^{120} possible moves, compared to 10^{80} seconds since the beginning of the universe.

One means of trying to overcome an incomplete knowledge of a domain is by application of heuristic rules. A heuristic rule is adopted from experience and it's basis may not be completely understood, such rules are commonly called rules-of-thumb and are said to be a frequent basis for expertise. Heuristic rules can be used to conduct a heuristic search which increases the likelihood of finding a solution by narrowing down the search space (Ernst and Newell, 1969).

3.3.2.6 Uncertainty

In many problem solving situations there is uncertainty over the value of one or more attributes pertaining to the solution. This is certainly the case in land evaluation where in the absence of firm information very subjective judgments have often to be made. One of the great benefits of expert system research has been seen as the development of methods for being precise about imprecision (Forsyth, 1984). A variety of methods for inexact reasoning have been proposed, but interest and development work has focussed upon three in particular : Fuzzy logic; Certainty factors and Bayesian logic.

a) Fuzzy Logic

Fuzzy logic was invented by Zadeh (1965) by an extension of classical Boolean logic, to real numbers. In Boolean algebra 1 represents truth and 0 is false and this is the same in fuzzy logic, but all fractions between zero and 1 can also be used to represent a partial value of truth. The AND, OR and NOT operators can be applied in a manner consistent with Boolean algebra, as shown below.

Where P_1 and P_2 are truth values for two different attributes:

P_1 AND P_2 = Minimum of P_1 and P_2

P_1 OR P_2 = Maximum of P_1 and P_2

NOT $P_1 = 1 - P_1$ (The inverse)

The aim of Fuzzy logic is the combination of uncertain evidence in a rigorous and consistent manner.

b) <u>Certainty Factors</u>

Shortliffe (1976) devised a method based on what were named 'Certainty Factors' (CF), for measuring the confidence that could be placed in a conclusion, on the basis of the evidence at any stage. A certainty factor is the difference between two computer measures:

$$CF[h:e] = MB[h:e] - MD[h:e]$$

Where CF[h:e] is the certainty of the hypothesis h given evidence e. MB[h:e] is a measure of belief in h given e, while MD[h:e] is a measure of disbelief in h given e.

CF's can range from -1 (completely false) to +1 (completely true) with fractional values between. MB's and MD's can range between 0 and 1. The system is not based on probability, rather the measures are a subjective grading. The operators AND, OR and NOT are applied in the same manner as for fuzzy logic.

Shortliffe also provided a method whereby new information could be combined in simple manner with previous results. The formula for MB's is:

$$MB[h:e_1,e_2] = MB[h:e_1] + MB[h:e_2] \times (1-MB[h:e_1])$$

The formula is the same for MD's and can be stated as: The effect of a second piece of evidence (e_2) on the hypothesis h given earlier evidence (e_1), is to move the fraction of the distance remaining towards certainty, indicated by the strength of the second piece of evidence.

The Shortliffe scheme provides for the possibility that inference rules as well as data may be uncertain. An 'attenuation' factor, representing the faith placed in the rule can, be 'tagged' to the rules and this is multiplied by the MB's before combining evidence, where more than one rule is concerned in the hypothesis. Theoretical justification for the Shortliffe method is questionable, but the empirical results in significant working systems such as MYCIN (Shortliffe, 1976), have focussed interest on the approach.

c) <u>Bayes Theory</u>

Bayes theorem allows the computation of relative likelihoods between competing hypotheses on the strength of the evidence. The 'Likelihood Ratio' (LR) is the probability of an event or evidence (E), given a particular hypothesis (H), divided by the probability of the evidence, given the falsity of that hypothesis (H') as in the formula:

$$LR(H:E) = P(E:H)/P(E:H')$$

The rule can be expressed in odds or probabilities. The Bayesian updating method can be expressed as:

$$O'(H)=O(H)xLR(H:E)$$

Where O(H) is the prior odds in favour of H and O'(H) is the resulting posterior odds given event E, as determined by the likelihood ratio.

d) The 'Uncertain' Controversy

Bramer (1985) considers the provision of an inference mechanism capable of dealing with uncertainty as essential to many domain problems. Forsyth (1984) lists the capability to reason with uncertain data as a distinctive feature of expert systems, but others disagree. White (1985) found that rule-based expert systems which have scope for inexact reasoning typically exhibit serious deficiencies of statistical inference. The principal deficiencies he demonstrated were: violation of the assumption of conditional independence; use of 'fuzzy logic' and difficulties arising from the use of subjective estimates of probabilities. White was particularly critical of the methods used in the PROSPECTOR system. He suggested giving up the goal of attempting to encode the knowledge of the expert directly into an expert system and favoured either the use of automatic induction for a logical representation, or statistical techniques if a statistical model is required. The SuperExpert and ExpertEase systems investigated here are examples of induction systems (section 4.4). The TIMM system (section 4.5) used in this study, is an example of an approach which incorporates statistical techniques into it's inference engine for representing a partial match.

The handling of uncertainty in expert systems is a research topic in itself. It is of importance to modelling of the land evaluation problem, since as already explained the procedures tend to be subjective and data is often lacking. Those using an evaluation often want evidence of it's reliability. Because of it's importance uncertainty has been considered in some of the applications software tested, but in the absence of agreement as to a reliable representation, the research has not focussed on this aspect of expert systems.

3.3.2.7 Induction Systems

It is in an effort to alleviate the knowledge gathering bottleneck mentioned in section 3.2.1 that researchers propound rule induction as a solution. Rule induction referred to here is the generation of rules automatically from examples. The process of induction is defined as general inference from particular instances (Coulson, 1969), in contrast to deductive reasoning which is the process of reasoning from the general to the particular. Facts can be derived from theories by deduction, whereas by induction, facts are used to arrive at theories. Shapiro (1987) describes induction as a type of informed conjecture based on the known domain of instances. Unless all relevant facts are available a theory may always be refuted when further facts come to light, so it is not possible to validate an induced rule in the same way as a fact can be proved by deduction. When concept learning is performed by a machine it is called inductive inference or computer induction.

Michalski & Chilausky (1980) showed that it is possible by automated inductive learning, to build a complete expert system from examples. The system built was for the diagnosis of disease in soyabeans. The conclusion was that computer induction techniques could offer a viable knowledge acquisition method if the problem domain was sufficiently simple and well defined. The Concept Learning System (CLS) of Hunt et al. (1966) was the first to use examples to automatically generate rules.

The ExpertEase and Super Expert systems used in this study, are examples of expert system shells using inductive inference to generate a rule from examples. The two systems are both based on an induction algorithm called ID3 (Iterative Dichotomizer Three), developed by Quinlan (1979) from the earlier work of Hunt et al. (1966). Quinlan's ID3 algorithm was extended by Paterson et al. (1982), into the Analog Concept Learning System (ACLS), which lies at the heart of the expert system shells used.

The ID3 algorithm takes objects of a known class, described in terms of a fixed collection of properties or attributes, and produces a decision tree over those attributes that correctly classifies all the given objects. The main advantage of using this particular algorithm is it's computational efficiency - as compared to other classification methods for the same task. It is well suited to using very large numbers of examples in forming the concept for a rule.

The rule-forming procedure will always work provided that there are not two objects belonging to different classes but having identical values for each attribute, in such cases the attributes are inadequate for the classification task and a 'clash' occurs

The inference mechanism minimises the number of tests required to classify an object. In developing ID3, it was assumed that the complexity of the decision tree, is strongly related to the amount of information conveyed about the class of an object by each attribute. In a sequence of attributes, the choice of which to test next is governed by which gives the most information towards discrimination between outcomes. Consequently some attributes may never be needed if others give sufficient information to predict the outcome.

The process of inducing a decision rule is described in the Super Expert User Manual (ITL, 1986) as follows:

1/ Create the rule node and associate all examples with it.

2/ If all examples at the current node are of the same class then stop.

3/ Choose the best attribute to split the examples at the current node. This is done according to an information theoretic measure described below. The type of attribute chosen may be logical or integer. If logical, then the tree splits at this point with one branch for each value of the chosen attribute. If integer, then a binary split is performed by partitioning the integer range about a threshold value. The threshold value is chosen to split optimally the examples at the current node.

4/ The current node has two or more branches. The examples at the current node are then associated with the branch nodes according to the value of the split attribute. Examples which have a "don't care" for that attribute are associated with all branch nodes.

5/ For each branch node repeat from step (2).

The choice of an attribute on which to split at any node is performed using an information theoretic measure based on the entropy of different states of the problem. The probability can be calculated of an attribute having a certain value for a particular outcome class. The entropy is calculated from the probability values, for each possible attribute value. By splitting an attribute the probability changes and consequently the entropy alters. The change in entropy is a measure of the usefulness of the attribute. The attribute with the highest value of entropy change is the next one on which to split. The mathematical representation of this process is explained by Paterson et al. (1982), and is given in appendix B.

3.4 Logic Programming and Prolog

3.4.1 Logic

Logic provides an alternative to the knowledge representations already discussed. Logic can provide a precise language to describe knowledge. Aristotle discovered the syllogism, thereby founding formal logic. Aristotle regarded logic as a useful tool in all enquiries, whatever their subject matter, (Ackrill, 1981). Logic provides the mechanism for deducing consequences from premises.

3.4.2 Computer Architecture

As with logic, computers require a precise and explicit statement of goals and assumptions, but expression of problems in a computer language has been constrained by technological limitations of the underlying machine architecture. Most modern computers are based on the Von Neumann concept (Augaurten, 1986). A program for a Von Neumann type machine consists of a sequence of simple instructions to perform operations on data held temporarily. An additional set of control instructions can affect the next instruction to be executed. The basic language used for these instructions is Assembly language. Assembly language is composed of a set of symbolic instruction codes. Inside the control processing unit (CPU) these codes are translated with the data into a binary form called machine language, so that the electronic circuiting can perform specific tasks. All other languages and operating systems are built from assembly language.

3.4.3 Tedium versus Intellectual Challenge

Abstractions from Assembly language were developed with a better formalism for programmers to express their intention in a more natural way, but Sterling and Shapiro (1986) argue that the so-called high level languages such as FORTRAN and PASCAL are still contrived to fit the Von Neumann architecture. The argument is that the Von Neumann characteristics have led to a separation of tasks in problem solving. There tends to be a first stage of thinking how to solve a problem and designing methods, followed by a second mundane and tedious task of translating designs into instructions for a computer. In this environment the explicit expression of knowledge in an acceptable formalism for the computer can be tedious.

In contrast to the Von Neumann environment Sterling and Shapiro (1986) contend that formalising knowledge in logic is often an intellectually rewarding activity and often adds insight to the problem under consideration. If thoughts can be organised as programs, a complex set of assumptions can investigated by 'running' the assumptions. The term 'rapid prototyping' has been coined, to mean that a conceptual solution to a problem should be developed as part of a working program that demonstrates it and exposes it's different aspects. This goal can be achieved through logic programming (Kowalski, 1983).

3.4.4 Logic Programming

A logic program is a set of axioms, or rules defining relationships between objects. A computation of a logic program is a deduction of consequences which is it's meaning. Kowalski (1983) showed that an axiom:

A if B_1 and B_2 and --------> and B_n

can be read, and executed, as a procedure of a recursive programming language, where A is the procedure head and the B_i's are it's body. To solve A, the B_i's will be solved in turn. The unification algorithm and resolution principle of Robinson (1965) can perform the basic data manipulation operations of variable assignment, parameter passing, data selection and data construction. Colmerauer (1973) developed a specialised theorem prover, written in FORTRAN, which embodied Kowalski's procedural interpretation of logical axioms and Robinson's resolution principle, it was called Prolog (Programmation en Logique).

A reduction is the basic computational step in logic programming. It corresponds to the repeated application of the rule of universal modus ponens i.e. 'if proposition P THEN proposition Q, P THEREFORE Q' (Hodges, 1978). P and Q are often compound propositions with logical connectives of 'AND', 'OR' and 'NOT'. So to prove Q true each individual proposition in P has to be proved true by reducing their component propositions. Q is implied by P and in Prolog Q would be the 'head' of a clause. Reduction is accomplished by unifying (matching) the head of a clause with a current goal and then resolving the sub-goals it is dependent upon.This involves backward chaining through a hierarchy of sub-goals, until the arguments to the predicate (relation) are only facts.

3.4.5 Prolog

Colmerauer's approach (1983) to the design of Prolog is similar to that adopted for knowledge representation in expert systems. Colmerauer views knowledge as a set of facts, specified by a set of rules; he represents each of the facts by a declarative sentence. The declarative sentence can be represented as a tree and trees were chosen as a data structure, because they are capable of expressing complex information and yet are simple enough to be handled algebraically and by a computer. Figure 3.6 from Colmerauer (1983) shows this structure for an arithmetic expression.

Formulae are used to represent tree patterns. The formulae are called terms and consist of "atoms" of information, variables, parentheses and commas. An atom of information is either a group of words, a number, or a special character.

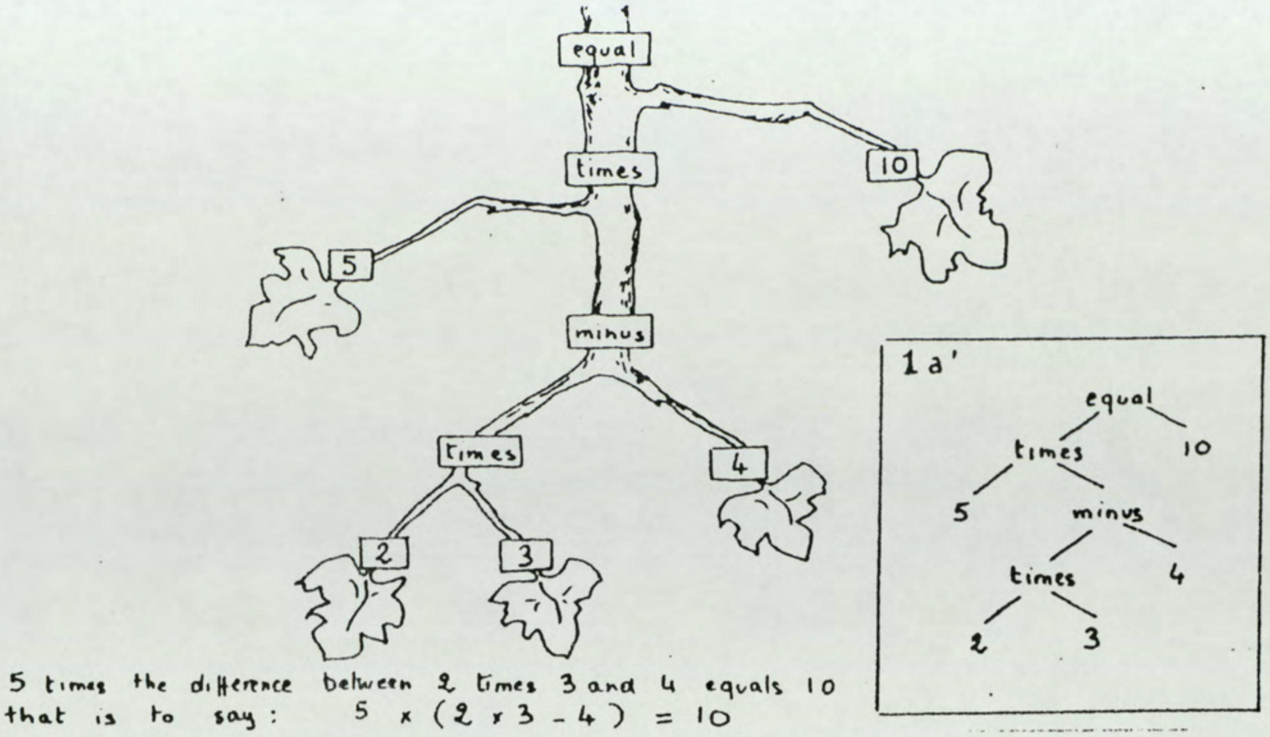

equal
times
10
5
minus
times
4
2
3
1a'
equal
times
10
5
minus
times
4
2
3
5 times the difference between 2 times 3 and 4 equals 10
that is to say: 5 x (2 x 3 - 4) = 10

Figure 3.6 An arithmetical problem represented as a hierarchical tree structure (after Colmerauer, 1983).

Two major decisions have to be taken in converting the abstract computation model for logic programs to a form suitable for a programming language (Sterling and Shapiro, 1986). First to be specified, is the arbitrary choice of which goal in the resolvent (current goal) to reduce. Then there must be a means of implementing the non-deterministic choice, of the clause from the program to effect the reduction. Prolog's execution mechanism is to choose the left-most goal rather than an arbitrary one, and replace the non-deterministic choice of a clause by sequential search for a unifiable clause, and back tracking to the next clause after each solution is investigated.

Provided the computation in a problem progresses sequentially, control in Prolog is similar to conventional procedural languages. The ordering of goals in the body of clauses corresponds to a sequencing of statements (Sterling and Shapiro, 1986). For instance A <---- B_1, B_2.....B_n, where A is the head of the clause and the goal to be invoked and the B_i's are sub-goals in the body of the clause, the clause can be viewed as a definition of procedure A:

```
Procedure A
      call $B_1$
      call $B_2$
      .
      .
      .
      call $B_n$
end.
```

The difference from a conventional language shows when backtracking occurs. In a conventional language, if a computation cannot proceed because all choices at a conditional break are inappropriate (e.g. all branches of a 'CASE' statement are false), then an error occurs. In Prolog the computation is simply retraced to the last choice made, and a different computation path is attempted.

Another major difference between Prolog and conventional languages is that it does not support destructive assignment where the contents of an initialised variable can change. Logical variables refer to individuals rather the memory locations.

3.5 Satellite Remote Sensing for Land Use Planning

The purpose of land use planning, is to optimise the use of land according to current theory and practice. A plan is normally made with the expectation of some changes in land use in the future. If the plan is complementary to a development programme in a developing country, then monitoring and evaluation of the implementation of the original land use plan will be an integral part of the process of land use planning. Land use planning is a dynamic process rather than a one-off exercise, representing a static situation.

In the context of land use planning for development, in order to justify change in land use, a thorough understanding of current land use practice must be shown. There are two main reasons for this: First, to identify where a change in practicе is required; second, to provide a baseline of information for comparison with the results of a change in land use. The baseline data is essential to any monitoring exercise. Random site visits alone could supply this kind of information, but there is a high risk of bias, and access is often a problem. Comprehensive ground survey takes a long time and the cost of such survey is also very high. Remote sensing may often provide a more effective and comprehensive means of identifying land cover and detecting changes in utilisation. There is still then a need for survey work to provide verification and additional socio-economic information, but the use of remote sensing may be an improvement on the methodology. Kannegieter (1985) suggested that remote sensing can make an important contribution to the rapid economic, efficient and accurate delineation of land units and determination of their qualities. This Kannegieter also acknowledges, is in addition to it's utility for inventory of present land use and monitoring. For the delineation of land units at the larger scales over relatively small areas, air photography rather than satellite imagery will usually be required.

Schultink and Lodwick (1981) in describing the CRIES project (Comprehensive Resource Inventory and Evaluation System) and evaluating the application of satellite imagery for land use studies, state that it provides an excellent data acquisition alternative for reconnaissance level or even semi-detailed surveys. They acknowledge the limitation of current landsat MSS data, which with it's low resolution, is inappropriate for many tropical areas, typified by small scale subsistence farming and inter-cropping. They conclude that in general, in many regions in the world where, comprehensive, multi-seasonal and timely aerial photography is incomplete, or non-existent, satellite imagery is an excellent alternative. Unfortunately it is just such a lack of support information, that makes validation and accuracy assessment of a satellite classification very difficult and thus questions the value of the results.

As long ago as 1972 Brooner and Nichols (1972) concluded:

> "There is growing agreement that the 'state-of-the-art' of remote sensing is technically advanced to the point of being very useful for the detection of environmental resource data. The need for more, better, and timely land use data for planning is with us"

and

> "The realization of systems to integrate data derived through remote sensing techniques with multiple sources of exogenous data and to manipulate, store, retrieve, display and update these, will enable the transfer of remote sensing applications and technology into the effective operations of land resource management".

Brooner (1981), contended that over the subsequent decade, many of the GIS design developments have been independent of the advances and developments in remote sensing, with consequent problems of integration and incompatibility. This problem of interfacing was also noted by Myers (1981). Brooner anticipated that the problem of data combination would become worse rather than better, because remotely sensed data would be so readily available that leaving it out of the database as had often been the case, would no longer be permissible.

With this background to the use of remote sensing in land use planning, it was considered essential to include remotely sensed data in the development of techniques for automating land use planning.

The discrimination of crops by satellite remote sensing techniques is a very difficult applications problem (Allan, 1987). The accuracy with which cover has been predicted has been poor or at least unsatisfactory for agricultural and environmental management purpose (ESA, 1986; INRA/CNES, 1982 and 1984). Significant improvements in technique have been made, such as the knowledge based and multi-date methods of Belward and Taylor (1986), which can improve classification accuracy by 10-20%. For U.K. crops this still meant a typical accuracy of 60%. With the more complex and smaller scale farming systems commonly found in the tropics this accuracy level would probably be lower.

Although considerable research effort has been invested in trying to find the most accurate techniques for image processing of satellite data to identify land cover, considerably less has been done to establish the benefits of integrating ancillary data. Such integration is usually in the context of an information system to provide the most contemporary and comprehensive resource inventory data. In other words there is an interface required between GIS and remote sensing technologies. Young and Green (1987) noted that the mutual, potential benefits of the two technologies have been discussed at length (Estes, 1984; Marble and Peuquet, 1983). Yet there appears to be a marked disparity between announced potential and operational realisation of this potential. Young and Green assert that both remote sensing and GIS have been technologically driven rather than developed to meet the needs of users. There have been only limited demonstrations of the benefits of remote sensing and Geographic Information Systems, in solving practical problems on an operational, cost-effective basis.

3.6 Geographic Information Systems

Land use planning has traditionally relied upon the production of one or more maps of the region under study, to convey the results of a land evaluation survey. It is usual to produce a map of current land use and then a series of maps showing the suitability of land for particular crops, often distinguishing between suitability at different management, or technology levels. The maps are normally accompanied by a lengthy report, often in two or three volumes. Currently in Thailand, much effort is applied to the production of such maps and reports at the reconnaissance level, for provincial planning. Experience elsewhere has shown that such careful work may well be under-utilised, either through lack of awareness, or an inability to readily manipulate the results, for application to a specific problem. Often the personnel making use of the survey are not land use planners and may not appreciate the degree of detail summarised in a land use rating, or they require more quantitative estimates of suitability.

A computer automated Geographic Information System (GIS) may be an answer to some of these problems. Different definitions exist as to exactly what comprises a GIS but the concept is quite simple. A GIS is a unique form of information system, in that data are indexed by using a geographic locator, which permits storage, retrieval and comparison of all data by spatial location. Any computer system that meets this definition can be termed a GIS. Some systems are much more elaborate and powerful than others and as with any software, the right system to choose varies with the particular application. Land Information System (LIS) or Integrated Geographic Information System (IGIS)are both synonymous terms in common usage. A possibly better term is Spatial

Information system (SIS). One of the most recent definitions was laid down by the Association of Geographic Information (1989), defining a GIS as:

> 'A system for handling data which is directly or indirectly spatially referenced to the Earth, It may be used for capturing, storing, validating, maintaining, manipulating, analyzing, displaying or managing such data. It is normally considered to involve a spatially referenced computer database and appropraite software. A primary function of a GIS is it's ability to integrate data from a variety of sources.'

GIS which have progressed roughly at the rate of advancement in computer technology, seek to capitalize on the synergism inherent in being able to automatically compare a variety of socioeconomic, environmental and land use data sets for the same point or area on the ground (Bryant & Zobrist, 1976). A GIS may simply be a manual system, or computer automated as is now more usually the case. Input data to such a system can be point, line, or spatial(i.e. coded data expressed by aerial coordinates and measurements), or facts and observations. Output consists of text, or tabular and graphic information.

There are a number of important advantages in using a Geo-referenced information system for land evaluation:

1. Archive data can be stored in a form which encourages utilisation.

2. Spatial data from maps and images can be encoded and stored in a form facilitating manipulation of the data.

3. Analysis can be achieved by integration of different spatial data sets by mathematical operations, using specific programs which form a part of the system, or other programs that can be easily interfaced.

4. Graphic displays can be shown of either the raw data, or of the results from mathematical operations and data combination. This is usually output as a thematic map.

GIS techniques offer advantages to the earth resource scientists in particular because the subject has traditionally involved reliance on map data and the production of new maps. In most countries soil research has long been recognised as of central importance to the improvement of agricultural productivity. Such research depends upon extensive soil surveying, involving the recording of very many characteristics for each soil profile pit, or auger core. Storage in a useful form of the multitudinous records that result from this activity, has always been a problem. Usually this has been accomplished by classification of the soil as being of a certain type, which is then recorded on a soil map. Inevitably this leads to a loss of site specific information on soil characteristics (Webster, 1977). Webster cites this problem as one of the reasons for adopting a soils information system.

The Soil Survey of England and Wales has for some time recognised the utility of an automatic soil and land classification (Rudeforth, 1975), although it has only recently managed to partially complete the task of establishing an information system (see section 2.2).

The Netherlands Soil Survey also has a GIS system which is a computer-based information system using earth science information from the Soil Survey Institute and the Geological Survey. It was particularly aimed at capturing with the minimum of transliteration archive information. To this end much use was made of optical character readers to input data directly to the system. At the heart of the system is the G-EXEC file organisation system, developed by the National Environmental Research

Council of the United Kingdom. G-EXEC includes a large number of application programs such as statistical packages. Another important feature of this system is that additional applications packages, such as the SYMAP mapping software (Sheehan, 1979), can be readily interfaced for input or output of information. Bie & Schelling point out that more powerful data handling allows new methodological research to progress. Also international soil data transfer is currently limited, due to lack of standardisation in classification and analytical methods. This has been recognised in the establishment of the International Board for Soil Research and Management (IBSRAM) program, within the headquarters of the Department of Land Development (DLD) in Bangkok.

Recently agriculturalists have become aware of the possibilities for gleaning useful information from remotely sensed images. This will need to be considered as an essential input to any future GIS. With improved resolution of satellite imagery, down to 20 metres for multi-band imagery from the SPOT satellite, and with more frequent imaging, useful applications in crop management and yield prediction are feasible (Jackson, 1984). Although the limited number of bands (3) with SPOT data and the correlation of two of the bands, diminish the value gained from it's more frequent periodicity and higher resolution.

The Lauragais project (Jeansoulin, 1983) to evaluate SPOT simulation data used an information system approach, where the manipulation of information to create a database in geometric form was crucial to the analysis for agricultural and yield assessment. For agricultural applications the potential for integration of remotely sensed data and other data, has been recognised and applied for some time in the U.S.A., exemplified in the LACIE project (Powers et al., 1979) for global crop forecasting. The operational systems are already providing forecasting information, such as CROPCAST (Merritt et al., 1981) and the CRIES project (Schultink & Lodwick, 1981). One problem has been the integration of different data units, and the first GIS tended to be of the polygon overlay or grid cell format, but not both: With satellite imagery geo-coding is in a raster format (Bryant & Zobrist, 1976). A polygon overlay technique represents geo-coded information as irregular geographic areas, simulating the known spatial distribution of the statistics. A grid cell technique imposes a digital grid over the spatial data, whereby geo-located data is retrieved through the cross tabulation of variables, encoded within a particular cell. A raster scan is equivalent to an ultra-fine mesh, grid cell data set. The system Bryant & Zobrist put forward, for integrating existing GIS data with thematic maps and satellite imagery, is based on the premise that geo-coded data can be referenced to the fine grid raster scan. The Image Based Information System (IBIS) they developed makes use of digital image processing techniques. Such an approach probably indicates the route to an inexpensive GIS of the future, well suited to agricultural applications.

Bryant and Zobrist also specify four basic criteria which GIS should satisfy if they are to be useful.

1/ They should provide specific point locations, as well as area locations of data.

2/ They should provide for variable aggregating (sub-setting) of the data.

3/ They should provide a method for representing spatial arrangements.

4/ They should be able to interface with mathematical and statistical programs which can be called as needed to aid in the analysis of spatially orientated data.

The last of these points is of particular importance in view of the recent developments in artificial intelligence techniques, particularly expert system technology. Interfacing of mathematical programs has to be easily accomplished, for the GIS to be utilised fully by resource planners and other users, and the need for a simple design, requiring no previous programming experience, has been noted by several authors (Campbell & Goldberg, 1981; Sinton,1979). Frank (1981) presents one novel and simple solution to the problem with the use of an electronic coordinate digitiser as the primary communication mechanism in a GIS system. With this approach, one compartment of the digitiser tablet is reserved for mounting a base map of the area, stored in a data base, while the other compartment contains an instruction menu. However, expert systems might fulfill this role of a "user friendly front end" in a more flexible and convenient manner.

Peuquet (1983) cites two major problems with interactive GIS. One is data storage difficulties and slow response times and the second is that they tend to be applicable to only a narrowly defined set of problems. The underlying cause of these problems is that GIS are characterised by: a tendency to become extremely large; fuzzy geographic boundaries tend to give incomplete, imprecise and error-prone data sets; and the number of possible spatial interrelationships is very large. Peuquet believes that the combination of artificial intelligence (AI) techniques with data base management systems and computer vision, shows great potential for dealing with these problems.

CHAPTER IV

METHODOLOGY

4.1 Introduction

As was explained in section 1.1 the aims of this work, were to investigate the application of knowledge based programming to land use planning; to show how the spatially related data involved could be processed such that it could be integrated with socio-economic information; and to show how IT tools might help in extending land evaluation beyond purely physical considerations, to include socio-economic data. Several different IT techniques were required to attempt to meet these aims, some proving more successful than others. For the sake of simplicity and to preserve the natural progression which occurred, the techniques are described in essentially the same order they were attempted, although the image processing work tended to be a recurring task, dependent on availability of a machine and the acquisition of satellite imagery.

4.2 Micro Expert Shell

4.2.1 Purpose

Micro Expert was designed as a microprocessor-based system in the anticipation of having an interactive system, which could be taken to a work site (Cox 1984). The shell was designed to emulate the reasoning process of an expert in a particular field. It enables the user to build and use an expert system based on the model provided by an expert. Micro Expert is based on the PROSPECTOR approach, originally developed to assist field geologists (Duda et al., 1978; and see section 3.3.2.2). It is written in PASCAL and is marketed by ISI Ltd. (a joint venture between ISIS systems Ltd. and ICI). It is known as an 'advice language' type system in that it offers advice on the basis of a rule set (production rules, as explained in section 3.3.2.4) supplied by a knowledge engineer. It is consulted interactively by the user who has a particular problem. The Micro Expert model, formulated in the purpose designed 'advice language' consists of a number of interacting hypotheses forming one or more tree structures. The 'leaf nodes' of the tree consist of questions which the system will ask the user, if that information is required. Both forward and backward chaining are used in the model (see section 3.3.2.5). The system chains back from the current goal to find the next question to be asked to justify the hypothesis. When an answer is given the new information is propagated through the network, updating the likelihood of all hypotheses effected, immediately. In principle Micro Expert seemed to be the type of expert system shell that a consultancy company might wish to use for a part of a land use assessment system.

4.2.2 Application Tested

This expert system shell was evaluated to determine whether the production system approach, together with a hierarchical control structure, was appropriate to automating the land evaluation task. The particular test problem chosen was selected from the Regional Soil Bulletin for the West Midlands (SSEW, 1984). The objective was a simple system to determine trafficability of land based on three parameters: Moisture Regime; Soil Depth; and Drainage Status. Trafficability is a measure of the capacity of land to withstand and permit the use of agricultural machinery.

4.2.3 Methodology

A Bayesian rule (see section 3.3.2.6) was designed to express the risk of poaching with each of the outcomes representing a different risk category. The risk was determined on the basis of certainty values designed to represent conventional categorical ratings e.g. S1, S2 and S3. Conjunction and disjunction were used to express the possible combinations of attribute values occurring, by use of 'AND' and 'OR' rules respectively.

In Micro Expert it is difficult to separate the control structure from the knowledge base, this is explained by the network underlying the system. The relationships (links) between facts (nodes) in the knowledge base, are used to build the hierarchical tree which forms the control structure. There are three major types of relationship between nodes in a model:

1) Rules to establish the likelihood of a hypotheses given the likelihood of antecedent hypotheses - using Bayesian logic.

2) Logical connectives e.g. AND, OR, NOT, EQUIV.

3) Contextual relationships allowing ordering, to avoid trying to assert a rule without the necessary information - Blocking.

Bayes theory has been used in Micro Expert to determine how prior odds of an hypothesis change, with a true observation by a "sufficiency factor" (LS) and by a "necessity factor" (LN) if the observation is false. A Bayesian rule provided in the advice language, allows the knowledge engineer to set the prior odds and the weightings of any relevant antecedent hypotheses.

The apriori odds of the Bayesian rule are multiplied by a factor for each of the antecedent hypotheses. If the certainty of an antecedent hypothesis is -5 then the factor is equal to the LN value. If the certainty of the antecedent hypothesis is 0.0 then the factor is equal to 1.0. If the certainty of the antecedent hypothesis is +5 then the factor is equal to the LS value. The relationships in a Bayesian rule are shown diagramatically in figure 4.1. Where 'Rule Name' is assumed to have an apriori probability of 0.1. In other words on average the particular rule is true in 1 out of 10 cases.

```
             FINAL CERTAINTY EXPRESSED AS PROBABILITY
                                |
                 _______________|_____________
                | RULE NAME                   |
                |               Prior         |
                |               Probability   |
                | BAYESIAN      0.1           |
                |_____________________________|
                                |
       LS 10            LS 100  |       LS 20
       LN 0.1           LN 0.01 |       LN 0.5
                                |
          Rule 1               Rule 2             Rule 3
```

LS = A Number representing the logical sufficiency factor, which is a weighting to express the truth of the antecedent hypothesis.

LN = A Number representing a logical necessity factor, which is a weighting to express the falsity of the antecedent hypothesis.

Fig 4.1 Illustrating a hypothetical example of a Bayesian rule structure in Micro Expert

Logical conjunction and disjunction are incorporated into the inference network by using fuzzy set theory as explained in section 3.3.2.6. For conjunction the logical connective AND is used and the probability of the result is defined as being the minimum of the conjuncts. Where the probability of a result is the disjunction of preceding hypotheses, the OR connective is used and the probability of the result is defined as the maximum among the antecedents.

For input of numeric values a user can be prompted with a 'NUMERIC QUESTION'. Two parameters are associated with such a question, a low value and a high value, for validation that the response falls between the two. Before the answer can be used in the system, the value must be converted to a probability value. This can be done using either a 'RANGE' or 'MODULUS' rule. With a 'RANGE' rule, the numeric value is compared with a 'low range value', at or below which the probability of the hypothesis is set to 0, and a 'high range value' at or above which the probability is set to 1.0. Between the values, probability is calculated by linear interpolation. With the 'MODULUS' rule, the numeric value is compared against a target value and if it is the same, the probability of the hypothesis is set to 1.0. A maximum deviation is also attached to the rule and beyond this the probability is set to zero. For intermediate values either side of the target, linear interpolation is used.

Contextual relations are necessary to encompass the situation where hypotheses cannot be considered in an arbitrary order. Since this is usually the case, the blocking option provided to achieve it is essential in a system of any size.

4.3 Farmer Interviews

A series of farm visits was planned, to investigate methods of knowledge acquisition and gather knowledge on farmer experience, for rule-based systems. The farm visits were made in the province of Rayong in Thailand, chosen as the study area for most of the land evaluation work. Knowledge acquisition was by means of informal interview and a pre-prepared questionnaire. The questionnaire was revised after the first field visit and both questionnaire formats are presented in appendix I. The questions asked in the final version were concerned mainly with the following factors: soil series; size of farm; depth to groundwater; age; experience; crop requirements; fertiliser and pesticide requirements; and marketing information. Three questions (4,5,and 6, in appendix I) were included, to try and elicit an explanation as to why certain crops, were or were not grown. Three field visits were made on separate occasions to Rayong province. The first was of just one week's duration, the second two visits were each of about 10 days. Over the whole period 52 questionnaires were completed.

4.4 Inductive Expert System Shells

One distinct group of expert system shells are the inductive models. As explained in section 3.3.2.7 inductive inference is the process of hypothesising a general rule from examples. Two similar inductive shells were used in this work, ExpertEase adn SuperExpert.

4.4.1 Expert Ease

4.4.1.1 Design Purpose

Expert Ease is an expert system designed for performing inductive inference. Designed by Intelligent terminals Ltd (ITL, 1984). The main purpose of the system, is to allow rules to be induced from a set of examples of the particular problem domain.

4.4.1.2 Application Tested

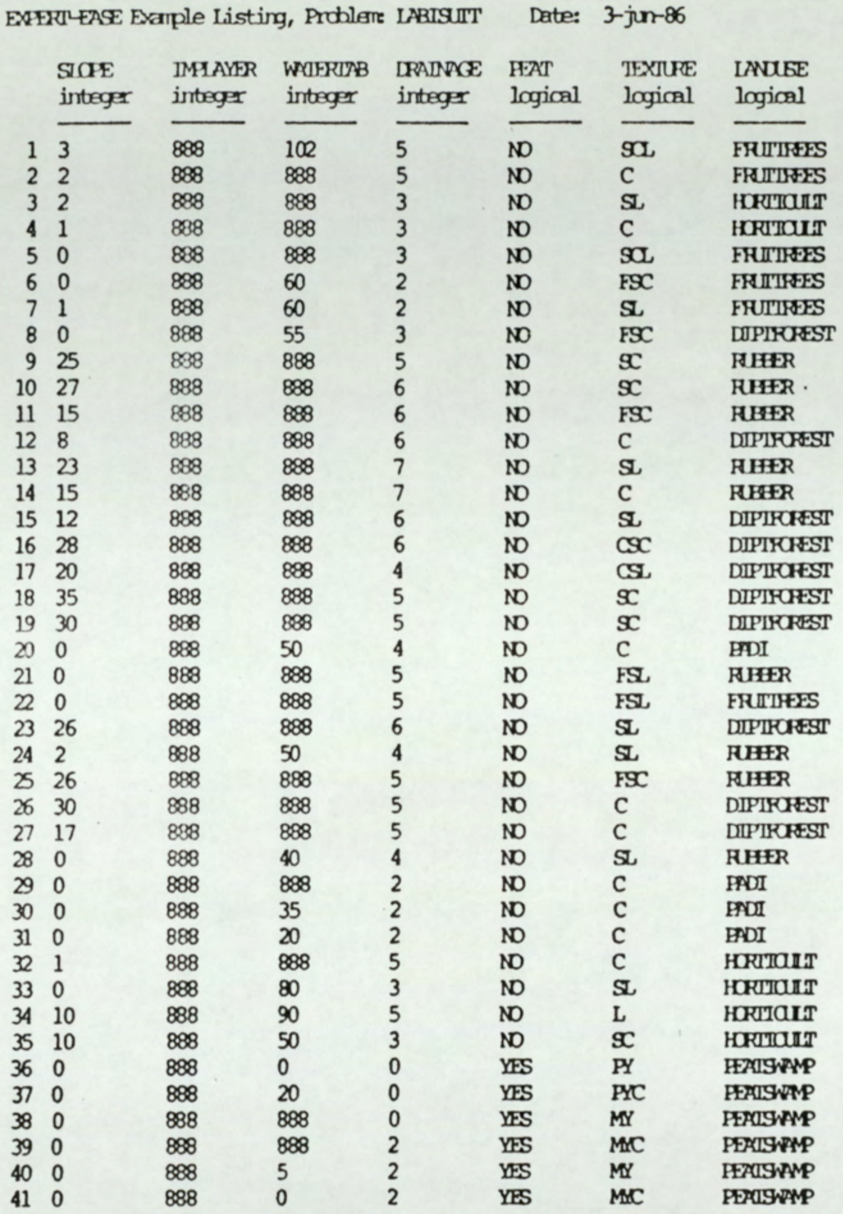
With the help of ULG staff, a case study from a Brunei land evaluation report was chosen to test Expert Ease. It was not easy to find a problem in the ULG archive, with sufficient examples available. This was in hindsight an indication of one of the shortcomings of this approach, for application in the ULG sphere of work (see section 7.3.1.6 page 212). The Brunei project had collected a large amount of data, on soil description and current land use, available in the form of coded sheets (see appendix B). The intention was to use Expert Ease to show which physical factors local farmers regarded as important in deciding on their choice of crop. Unfortunately there was no information on management or socio-economic factors, but as a preliminary trial to establish the viability of an inductive approach it was a suitable study.

4.4.1.3 Methodology

On inspection of the coded sheets it transpired that most of the area was in fact peat swamp or forest, but forty one examples covering five different land use types were selected (see fig 4.2). Initially six attributes were identified as being of potential importance in deciding on land use. Selection of attributes was constrained by limited availability of data, as although many attributes were provided for on the pro-forma sheets for soil description, most boxes were not completed and the recording of attributes was inconsistent, this is typical of the nature of data from such surveys.

The interface design of the Expert Ease shell is based on a spreadsheet style presentation with which most people quickly become familiar. There are four main screens that

Figure 4.2 Showing the example listing for the Brunei Land evaluation Report.



```
EXPERT-EASE Example Listing, Problem: LABISUIT     Date:  3-jun-86

     SLOPE      IMPLAYER   WATERTAB   DRAINAGE   PEAT      TEXTURE   LANDUSE
     integer    integer    integer    integer    logical   logical   logical
     -------    -------    -------    -------    -------   -------   -------
  1  3          888        102        5          NO        SCL       FRUITREES
  2  2          888        888        5          NO        C         FRUITREES
  3  2          888        888        3          NO        SL        HORTICULT
  4  1          888        888        3          NO        C         HORTICULT
  5  0          888        888        3          NO        SCL       FRUITREES
  6  0          888        60         2          NO        FSC       FRUITREES
  7  1          888        60         2          NO        SL        FRUITREES
  8  0          888        55         3          NO        FSC       DIPTFOREST
  9  25         888        888        5          NO        SC        RUBBER
 10  27         888        888        6          NO        SC        RUBBER
 11  15         888        888        6          NO        FSC       RUBBER
 12  8          888        888        6          NO        C         DIPTFOREST
 13  23         888        888        7          NO        SL        RUBBER
 14  15         888        888        7          NO        C         RUBBER
 15  12         888        888        6          NO        SL        DIPTFOREST
 16  28         888        888        6          NO        CSC       DIPTFOREST
 17  20         888        888        4          NO        CSL       DIPTFOREST
 18  35         888        888        5          NO        SC        DIPTFOREST
 19  30         888        888        5          NO        SC        DIPTFOREST
 20  0          888        50         4          NO        C         PADI
 21  0          888        888        5          NO        FSL       RUBBER
 22  0          888        888        5          NO        FSL       FRUITREES
 23  26         888        888        6          NO        SL        DIPTFOREST
 24  2          888        50         4          NO        SL        RUBBER
 25  26         888        888        5          NO        FSC       RUBBER
 26  30         888        888        5          NO        C         DIPTFOREST
 27  17         888        888        5          NO        C         DIPTFOREST
 28  0          888        40         4          NO        SL        RUBBER
 29  0          888        888        2          NO        C         PADI
 30  0          888        35         2          NO        C         PADI
 31  0          888        20         2          NO        C         PADI
 32  1          888        888        5          NO        C         HORTICULT
 33  0          888        80         3          NO        SL        HORTICULT
 34  10         888        90         5          NO        L         HORTICULT
 35  10         888        50         3          NO        SC        HORTICULT
 36  0          888        0          0          YES       PY        PEATSWAMP
 37  0          888        20         0          YES       PYC       PEATSWAMP
 38  0          888        888        0          YES       MY        PEATSWAMP
 39  0          888        888        2          YES       MYC       PEATSWAMP
 40  0          888        5          2          YES       MY        PEATSWAMP
 41  0          888        0          2          YES       MYC       PEATSWAMP
```

can be displayed within the system, each being used at a different stage in the process of building an expert system. The structure of the system, including the variables to be used and allowable values, is designed on the Attribute screen. The user then switches to the Example screen and enters the examples available. After induction of a rule this can be viewed on the Rule screen. An enquiry system, to interrogate the expert system built, can be designed on the Query screen. Examples for each of these stages are given in appendix B.

4.4.2 SuperExpert

4.4.2.1 Design Purpose

The SuperExpert shell was developed from the Expert Ease Shell already described . This was basically a later product from Intelligent Terminals Ltd., addressing some of the deficiencies in it's earlier product Expert Ease. The interface was improved, but more important the facility to introduce external data was added. Both systems are based on Quinlan's (1979) ID3 algorithm (see appendix B). As with Expert Ease, SuperExpert can either be used to induce rules from examples before building a query system, or rules can be entered as examples, with the system simply used as a framework to build a domain-specific expert system.

4.4.2.2 Application Tested

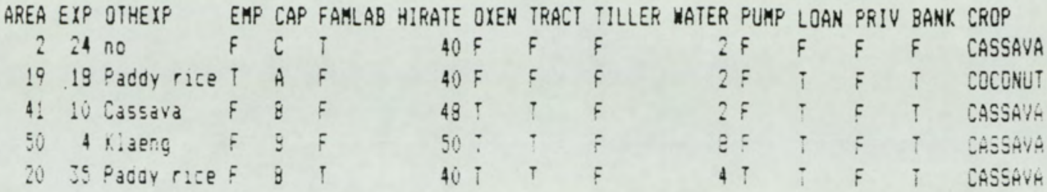
The SuperExpert shell was tried on data collected personally in Thailand. From the fieldwork in Thailand, by personal visits in the Eastern province of Rayong, examples were available recording certain land use factors and crops grown on a farm by farm basis. Similar examples were also available for a province in NE Thailand. The data for the NE was actually collected by DLD, on a questionnaire design arrived at as part of the work by the author in Thailand. The aim was to collect data in a form which was relevant to automation of the evaluation process. An entry scheme was designed in dBASE III to match the written questionnaire used. From this work a database of 617 records, containing information on land use attributes and the crop grown, was available for Yasothon province in North East Thailand. This provided an opportunity to test the hypothesis of using an inductive expert system, to elicit knowledge from farmers on crop selection, from a relatively large sample. The concept to be examined was that if sufficient examples could be gathered to accurately represent the farmer experience over an agro-ecological region, an inductive expert system could be used to derive a rule to determine what farmers were likely to grow. As argued in section 1.2 the farmers practice and heuristic knowledge is particularly important, so the ability to capture and express such knowledge was a priority. This method of using an inductive system was to be compared and contrasted with another technique using Prolog (see section 4.7).

4.4.2.3 Methodology

The data for the Eastern province of Rayong was collected in greater detail than that for Yasothon, but it was not possible to make a large number of visits. The Rayong data was also backed up by detailed notes on farm visits, as well as the completion of set questionnaires.

The data collected personally in Rayong was tested using the process of entering examples from a dBASE III file. The Rayong situation was well understood but the information was still incomplete and the example set was small, with only a total of 55 examples, derived from about half of the questionnaires collected. The other questionnaires were used to check the results. Figure 4.3 shows part of the example set with a brief explanation of the 20 attributes used, the full example listing is given in appendix B.

Figure 4.3 Screen dump showing part of the example set for the Rayong problem.



```
AREA EXP OTHEXP      EMP CAP FAMLAB HIRATE OXEN TRACT TILLER WATER PUMP LOAN PRIV BANK CROP
   2  24 no          F   C   T          40 F    F     F          2 F    F    F    F    CASSAVA
  19  18 Paddy rice  T   A   F          40 F    F     F          2 F    T    F    T    COCONUT
  41  10 Cassava     F   B   F          48 T    T     F          2 F    T    F    T    CASSAVA
  50   4 Klaeng      F   B   F          50 T    T     F          8 F    T    F    T    CASSAVA
  20  35 Paddy rice  F   B   T          40 T    T     F          4 T    T    F    T    CASSAVA
```

Key :-

<u>Field name</u>	<u>Meaning</u>
AREA	Size of holding in Rai.
EXP	Experience - Years farming.
OTHEXP	Previous crops grown or district farmed.
EMP	Other means of income [True(T) or False(F)].
CAP	Classification of wealth (see appendix B).
FAMLAB	Reliant chiefly (>50%) on family labour.
HIRATE	Charge for hired labour (Baht/day).
OXEN	Draught animals used(T or F)
TRACT	Tractor used (T or F).
TILLER	Power tiller used (T or F).
WATER	Depth to water table (m.).
PUMP	Water pump owned (T or F).
LOAN	Borrowing money (T or F).
PRIV	If loan is T, Is it a private loan? (T or F).
BANK	If loan is T, Is it a bank loan (T or F).
CROP	Type of crop grown.

The examples were prepared in dBASE III, requiring some type conversions and aggregation of attribute values to suit the format of SuperExpert. The text file was then sent to an ASCII text file. This text file was then loaded into the examples screen of the expert system designed for the problem.

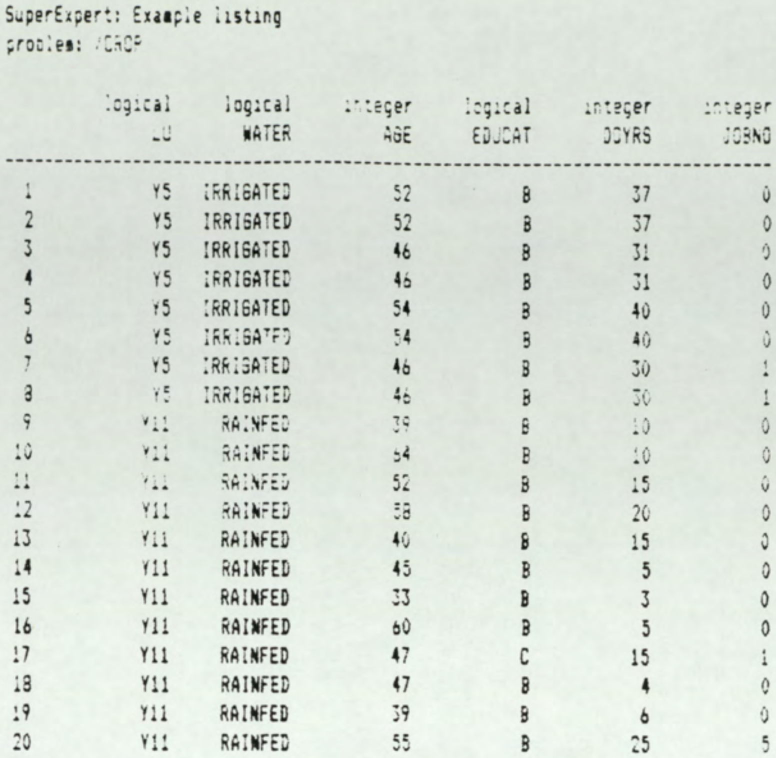
For the North East (NE) example, data from the 617 records available in a dBASE III file, 289 were chosen representing approximately half the examples for each crop. There were twenty different cropping systems identified as possible outcomes. This was a system limitation, as in fact more than twenty crop types had been recorded because of crop variety differences. Some varieties had to be grouped to accommodate the limit. A screen dump showing part of the format of the examples screen is given in fig 4.4.

Despite the design of a fairly comprehensive questionnaire, many of the questions were not completed by interviewers. This was probably due to the insistence of some involved in the questionnaire design, to try and collect data on everything that might possibly be of use. The indication was that the final questionnaire was too long and consequently cooperation from farmers was probably poor. Thus the information on land use attributes was incomplete. The variables covered are listed below:

1) Land Unit Name
2) Water Regime (Rainfed or Irrigated)
3) Age of Farmer
4) Level of Education of Farmer
5) Number of Years farming
6) Number of other jobs held by family
7) Tenure Status
8) Size of holding

Factor 1, Land Unit Name, summarizes many physical factors relating to the land, including soil and physiographic conditions. For a system to function correctly in deciding the crop, the land unit would be expected to be determined first, as socio-economic factors are irrelevant if physical conditions are limiting.

Figure 4.4 Showing part of the Examples screen for the Yasothon problem



```
SuperExpert: Example listing
problem: /CROP

             logical    logical    integer    logical    integer    integer
                  LU      WATER        AGE     EDUCAT      DOYRS      JOBNO
-------------------------------------------------------------------------
   1              Y5  IRRIGATED         52          B         37          0
   2              Y5  IRRIGATED         52          B         37          0
   3              Y5  IRRIGATED         46          B         31          0
   4              Y5  IRRIGATED         46          B         31          0
   5              Y5  IRRIGATED         54          B         40          0
   6              Y5  IRRIGATED         54          B         40          0
   7              Y5  IRRIGATED         46          B         30          1
   8              Y5  IRRIGATED         46          B         30          1
   9             Y11    RAINFED         39          B         10          0
  10             Y11    RAINFED         64          B         10          0
  11             Y11    RAINFED         52          B         15          0
  12             Y11    RAINFED         58          B         20          0
  13             Y11    RAINFED         40          B         15          0
  14             Y11    RAINFED         45          B          5          0
  15             Y11    RAINFED         33          B          3          0
  16             Y11    RAINFED         60          B          5          0
  17             Y11    RAINFED         47          C         15          1
  18             Y11    RAINFED         47          B          4          0
  19             Y11    RAINFED         39          B          6          0
  20             Y11    RAINFED         55          B         25          5
```

Key :-

Field Name	Meaning
LU	Land Unit
WATER	Irrigated or Rainfed
AGE	Age of farmer
EDUCAT	Level of Education
DOYRS	Number of years farming
JOBNO	Number of other jobs

4.5 The TIMM Expert System Shell

4.5.1 Design Purpose

Timm (The Intelligent Machine Model), is a computer program for building expert systems. It is an interactive tool that claims to perform the function of the knowledge engineer by querying the expert, constructing, and interpreting the knowledge base. No specialized knowledge engineering support or dedicated personnel are supposed to be required. A pattern-directed problem solving method is adopted. The analogical, partial match inference engine in

TIMM looks for symbolic analogies to a situation, among groups of similar rules in an expert system knowledge base. The rules may be an incomplete set of examples gathered, or an established and complete rule set.

4.5.2 Application Tested

A system was designed, to predict what most farmers would grow, given certain conditions. The "Farmer Decision" system was based on field interviews from the second field visit. This system was built in an effort to develop a working model quickly for demonstration in Thailand. Although examples from interviews were used, they were treated as established rules, rather than as instances for induction.

4.5.3 Methodolgy

The questionnaires used to build the TIMM system, were the same as the set used in the SuperExpert Rayong induction system (see appendix B). The important attributes had been identified from the farmer interviews. The 'BUILD' function in TIMM was selected and each attribute was listed as a factor influencing the decision as to which crop to grow. Possible crop choices were also listed. The nature of the factors also had to be specified i.e. whether unordered; linearly ordered or circularly ordered and whether they were numbers or phrases. The system was then trained by defining rules of the form:

'If <factor> is <factor value> Then <choice>'

as shown in figure 4.5. A primary knowledge base of such rules was defined, by reference to the questionnaires. The option to create a secondary knowledge base of impossible situations was not implemented, on the basis that such information was not available.

```
Rule 1
------
   If:
     LONG TENURE           IS YES
     CAPITAL               IS 2
     OTHER EMPLOYMENT      IS NO
     OTHER BUSINESS        IS NO
     LONG TERM             IS YES
     LABOUR SHORT          IS NO
     SUGAR FACTORY         IS 99
     PINEAPPLE FACTORY     IS 99
     SIZE                  IS 14
     SLOPE                 IS B
     WT                    IS 6
     FLOODING              IS NO
     SOIL                  IS 3
  Then:
     CROP CHOICE           IS RUBBER(100)
```

Figure 4.5 Showing the format in which rules for the TIMM knowledge base are expressed.

A certainty factor can be indicated for the crop choice, in brackets, in this case reflecting the percentage of the crop area in relation to total holding. Where an instance represents a partial match with several rules, the certainty of each rule is combined by averaging the certainty for each crop in every rule, dividing by the number of rules concerned.

4.6 Relational Databases - dBASE III

4.6.1 Design Purpose

The dBASE III software (Ashton-Tate, 1984) is a relational database management system providing file handling consistent with the theories of relational analysis as described in 3.1, and including an applications language for programming. The data files can be extended without effect on the programs.

4.6.2 Application tested

The supposed advantage of employing relational analysis was that relations would behave in a well defined way so that changes in the data structure, such as addition of new attributes, would not necessitate major rewriting of the program. Additional data records can also be added easily to existing databases.

The intention was to model the 'core' task of a land evaluation system, whereby a suitability value is derived from the values of several attributes. Further, it was intended to examine how easily the analysis results could be interfaced to other possible components of a LIS.

4.6.3 Methodology

Data was available on requirements of commonly grown crops in Rayong province, Thailand. The sources of data were: international guidelines (ILACO, 1981); local publications; DLD reports and local knowledge. The information on land qualities used was based on DLD guidelines (see appendix I). The dBASE III system written was then interfaced with a spreadsheet macro written in LOTUS 123 to obtain a Financial index from the suitability rating.

4.7 An Artificial Intelligence Language - Prolog

4.7.1 Design Purpose

Prolog is a programming language well suited to the implementation of artificial intelligence applications. Artificial intelligence research revolves around language understanding, where typically symbols are manipulated rather than numeric data. In Prolog the data structure is the logical term, which at the highest level is in the form of a rule, expressing the relationship between different data items. The sub-structure of such a term can be broken down to variables and facts. The focus on a structure based on logical terms means that it is knowledge which is manipulated in Prolog, rather than simply data. Manipulation of the logical terms, is accomplished by the reduction process based on a unification algorithm, as explained in section 3.4.4, page 45. Procedural control does not have to be exercised, as the unification rule ensures that the relevant clauses (logical terms) will be resolved to fulfil the ultimate goal. So in Prolog the program is essentially the data, exressed as relationships in a knowledge base. All alternative solutions to a proposition will be investigated by backtracking, unless control is specifically enforced. It is for this reason that Prolog is particularly suited to problems where a recursive model applies and all possible solutions are of interest. Prolog is also expounded as an important tool for 'fast prototyping', whereby initial ideas can be implemented in a program very quickly, and the logic tested at an early stage.

4.7.2 Application Tested

The intention was to investigate the application of Prolog to design of an information system, for all aspects where Prolog might have some natural advantages. In particular the potential for 'fast prototyping' was investigated, through construction of a crop selection system for Rayong province in Thailand, based on farmer interview responses. This was to be compared with a similar approach to crop selection using induction shells and TIMM. The possibility of using Prolog as a knowledge acquisition tool at the interview stage was also investigated. The graphic display and user interface facilities within the Turbo Prolog System were implemented in some of the programs written, to assess the advantages and disadvantages of the Prolog interface over other more traditional systems and software.

4.7.3 Methodology

Answers to pre-defined questions in the farmer interviews were used to construct an inductive expert system, but for the work on prolog the more subjective information gathered ad hoc was used. Some of the questions (namely 4,5 and 6, see APPENDIX I) were designed to try and elicit the farmers opinior on which crops should be grown under certain conditions. Together with notes taken at the time of interview, this information was analysed and rules were derived by inspection to show which factors were pre-requisites for choosing to grow certain crops. All the main crops encountered in the field survey were covered.

The derived rules were then programmed in Prolog and a query system was designed on an interactive 'Question and Answer' basis.

4.8 Satellite Remote Sensing

4.8.1 Design Purpose

As explained in section 1.3 of the introduction, digital image processing currently tends to be considered immediately in almost all land use planning exercises in developing countries and is often cited as an important component of GIS projects. Yet the effectiveness of the approach and understanding of what can be achieved is still not clearly established, as was evidenced in this work (see chapter 6). The objective in using a microcomputer-based Digital Image Processing System (DIPS), was to establish the feasibility of implementing such techniques in a real consultancy environment and to consider how the results from remote sensing could be combined in a micro-based Land Information System (LIS), with some of the other components investigated, particularly the socio-economic analysis.

4.8.2 Methodology

The techniques of digital image processing were used in two separate phases of this study. Initially for the province of Rayong in Thailand, in considering the resource information and other data required for land evaluation, Landsat MSS uata was processed to obtain a land cover classification. The results of a minimum distance classification were then combined with soils information to try and improve the classification. These results were then compared with a map of current land use, drawn up by the DLD from information derived by field survey and air photographs, contemporary to the Landsat imagery. Later image processing was used in a real situation, where a client of ULG required resource information from remote sensing and wanted to derive a land use plan, largely on the basis of satellite imagery. To some extent the client misunderstood the capability of the technology, but as a case study it provided an actual example of the utility and cost effectiveness of the approach, compared to other options. The detailed description of the methodology for each of these exercises follows.

4.8.2.1 Rayong case study

The province of Rayong lies on the Eastern Seaboard of the Gulf of Thailand. It covers an area of approximately 3,552 Km^2 and is one of 73 provinces in Thailand, the location is shown in Fig. 4.6. Table 4.1 summarizes the agricultural statistics relating to the province (OAE, 1986). Fieldwork centered around the district of Ban Khai as indicated in Fig 4.6 .

Table 4.1 Summarizing the land utilisation in Rayong province, Thailand (1986 figures).

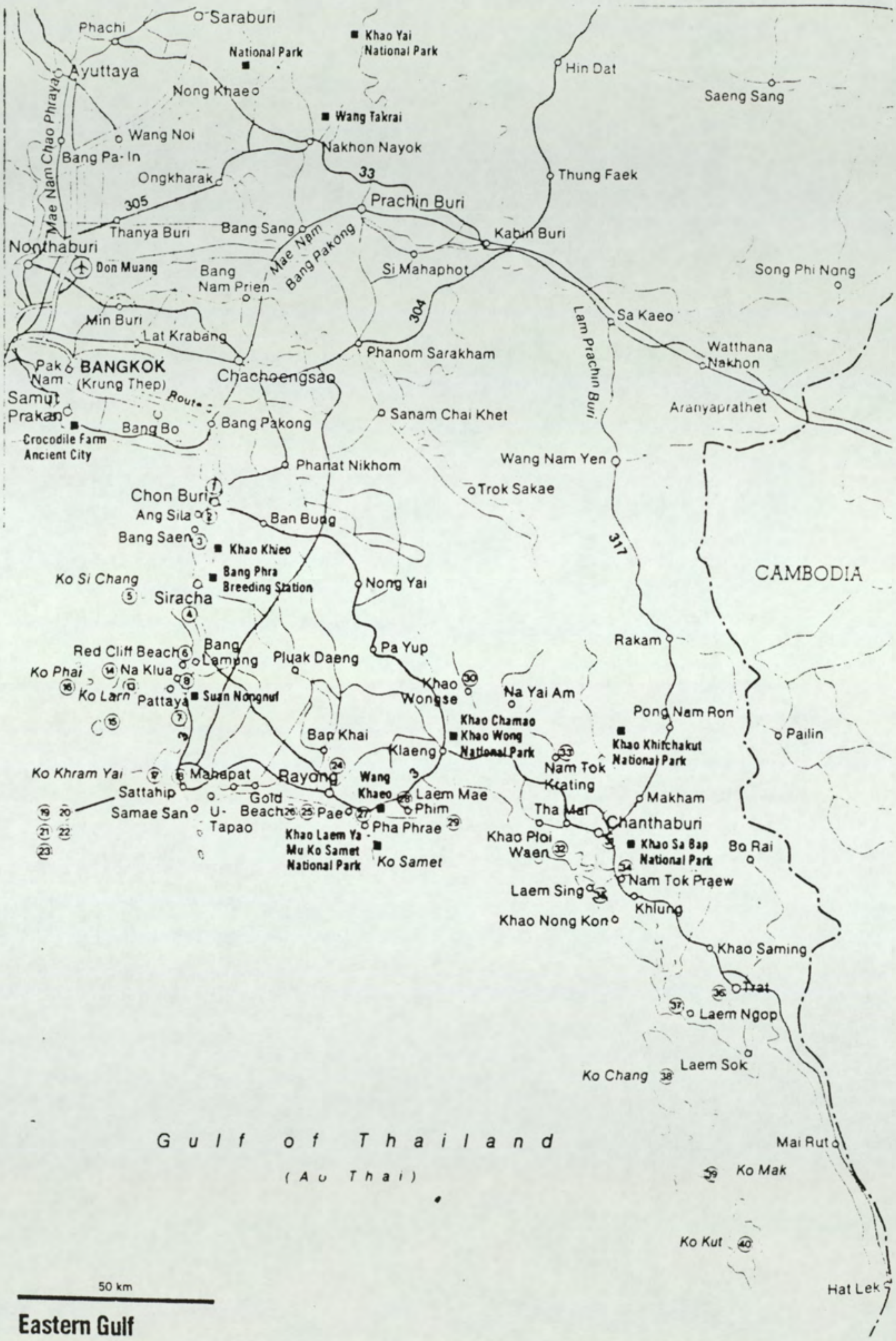
Land Utilisation in Km^2(%)

Total	Forest	Paddy	Field crops	Tree crops	Vegetables	Grass	Other
3552	241	296	1342	348	3	13	1309
(100)	(6.8)	(8.3)	(37.8)	(9.8)	-	-	(36.9)

This particular province was chosen as a case study on automation for land evaluation, at the advice of the DLD in Bangkok. It was convenient to the DLD in that they were collecting data on the province at that time, so provision of support in terms of transport and a translator was justified. More important though, the agriculture of Rayong province is very diversified, thus providing an interesting and broad example of the conditions under which a land use evaluation is actually made.

Rayong is a comparatively wealthy province with a substantial base of plantation agriculture. The main plantation crops are rubber; durian; rambutan; coffee; coconut; pineapple and sugar cane. Other crops grown are: cassava; paddy rice; groundnuts; mangos; bananas; tamarind; mungbean and vegetables. A complete list is given in appendix F. Inter-cropping is common, especially between newly planted plantation trees such as rubber or durian. Most of the land is in the hands of private ownership, with some families owning large tracts of land, but there are also many small landowners and there have been several settlement schemes, usually allocating 50 rai (8 ha) per family.

Figure 4.6 Showing the location of Rayong province and the study area of Amphoe Ban Khai.



Rapid changes were taking place in the cropping systems in Rayong at the time of the Fieldwork. Over the previous 10 years cassava had been widely planted where previously it was never grown before. This was in response to generous EEC contracts with the Thai government. At the time of the fieldwork cassava still covered the greatest proportion of Rayong province, but government policy and extension advice had changed. The danger of a price collapse was resulting in moves to diversification and the widespread problems of soil erosion and it's consequences were finally being recognised. For some time rubber planting had been encouraged, through the Rubber Research Institute and incentives of soft loans and supply of inputs were still being offered. Prices for durian fruit had recently soared and because of the healthy profits to be made in that crop, many people were planting durian.

The department of agriculture tries to keep track of such changes in cropping system, by sending out officers each year, to the village headman and master farmers, to ask them what they are growing. The statistics are questionable, because of problems such as tax evasion and they convey no impression of the spatial distribution over the province. It was obvious that information on current land use from the Department of Agriculture was not adequate.

Information on current land use was available from a section of DLD, although this was compiled on 1981 data mainly from air photographs and field survey, with some extension of results by inspection of landsat MSS images on an additive viewer.

Without information on current land use it would be impossible for planners to take account of the volatile situation, the only solution seemed to be to use remote sensing. Thus the intention was to test the viability of using satellite data to provide information on land use at the provincial level. Current information contemporary with the ground survey was obtained but there were only sufficient funds to purchase 9" positives in bands 7 and 5 for Landsat 5 MSS. Later from a UK source an MSS CCT was obtained free of charge for a date contemporary with the DLD land use survey. This good fortune allowed the accuracy of some of the processing to be checked. Both sets of data were processed. Techniques for handling the positives were of interest in themselves, as these images had to be digitised before analysis could be undertaken. Under normal circumstances data would not be handled in this form, but since much map data and other images are only available in hardcopy, techniques for digitisation are important and relevant to the viability and costing of such an exercise.

Processing of the satellite data was undertaken on an IBM-AT system as described in appendix F (Flach, 1987). The CCT of Rayong from December 1981 was loaded onto a DEC VAX cluster. Subsampled scenes and extracts were then sent across to the IBM via a serial line, using the Kermit communications program. Each 512 x 512 extract was stored in all four bands on high density floppy diskettes.

The first step in processing the MSS data was destriping to remove sensor effects and then geo-referencing of the images to map coordinates. Since soil and topographic maps were to be integrated with the imagery, it was decided it would be best to reference all to a common grid. A 2D transformation (Foley & Van Dam, 1982), incorporated in a C program, which was part of the Aston Software Suite (see appendix F), was used to accomplish the geometrical transformation. The precise transformation calculated was approximated by a least squares fit. Higher order polynomial transformation can be used but, for all the integration of map data done as part of this work, this approximation of the translation, scaling and rotation required, was found to be adequate.

Six points well spaced over the MSS extract were used, being easily identified for comparison with the same control points from the topographic map. The two sets of coordinates for the same positions where then input into the transform, which took about twenty minutes to run on the 512 x 512 scene. Later this was reduced to about 20 seconds.

Classification on bands 4,5 and 7 of the imagery was by means of a minimum distance algorithm. The choice of algorithm was in part decided by the capabilities of the system used, in that at the time a maximum likelihood algorithm was not available on the micro-based system. Later, the maximum likelihood program became available and when used it did provide a slightly higher accuracy, but for the integration of data sets the minimum distance classification was used. Under the minimum distance algorithm, pixels are assigned to a class on the basis of the minimum euclidean distance to class means from the pixel reflectance values.

To establish whether additional ancillary information improved the information on current land use, soil map data was integrated with the classification. The DLD map of current land use also had to be digitised, to allow estimates of accuracy to be made. The process of digitisation was in itself of interest as this can be a major constraint to use of a low cost image processing system on a project overseas. There were several available options as listed below:

1. Using a digitising tablet
2. Frame-grabbing
3. Automatic scanning

Using a digitising tablet is a standard method, but it is slow and tedious. Observations on other work using this method showed that for the type of maps concerned and the area to be covered digitisation would take about half a day for each map. The intention was to investigate new techniques so methods 2 and 3 were chosen for evaluation.

The Frame-grabbing method simply converts hardcopy data such as a map into a raster format digital file. Further processing is usually required to extract the necessary information, as the pixel values do not have an inherent meaning. The subsequent feature extraction may be either interactive or automatic. Both methods were tested for the video-grabbed data and for the scanned data. Data was automatically scanned using a drum-scanning densitometer with the cooperation of the JOYCE-LOEBL company based in Gateshead.

For frame-grabbing a video camera is attached to one of the image processing cards on the micro-system, via a cable transmitting a standard RS-170 or CCIR RGB video output signal (these are industry standard signals composed of analog video information and timing information). One frame was digitised through an analog to digital converter on the image processing board. One frame comprises the horizontal scan lines of a complete image on a CRT, from the continuous video image (maintained by 30 frames/second). Distortions and other aberrations in the data do arise in video grabbing, particularly vignetting and other forms of interference. Interaction between the lens and the filter also introduces some optical interference effects, when grabbing on one image processing board, using different filters (Red, Green and Blue) to allow a full colour image to be composed. The quality of image produced by automatic scanning can be of a much higher standard.

Automatic scanning was by means of a drum scanning densitometer. Hardcopy was mounted on a rotating transparent perspex drum and the density over the image was measured. Density of the image was measured through a lens, to a set of four photomultiplier tubes, collecting the light reflected from an incident light flux of white light, from a tungsten-halogen source, giving a broad band density measure (Scarpace, 1978). In contrast to the video digitisation where several 'grabbed' images had to be referenced together, even large maps can be easily digitised to a single file on a drum scanning densitometer. The amount of data varies with the aperture set and it was possible to go down to 12.5 um. The resolution of the emulsioi on the positive transparencies was about 25 um so the aperture was set to that level for those images. For the soil map an aperture of 100 um proved adequate.

An attempt was made to automatically extract the soil boundaries from the digitised soil map and so produce an image with boundaries at a distinct reflectance level from the background. Automatic feature extraction offers the possibility of a considerable time saving, so various techniques were tried. The soil boundaries and codes on the original map were marked in magenta, against an overall grey background.

A sequence of 'grabs' were made of the soil map using different filters and without any filters. The filters used were: Red; Green; Blue; Cyan; Yellow and Magenta. Different combinations were manipulated by subtraction and ratio, to try and isolate the soil boundaries to a distinct range which could then be thresholded to the desired value.

Where automatic extraction was not successful boundary lines were vectorised interactively by following the lines displayed on the monitor with a cursor directed by a 'mouse' device.

Once boundary lines had been isolated standard fill routines, supplied with the image boards, were used to fill within a boundary to a particular colour, representative of a certain soil code.

The introduction of soil information to the classification was not bound to improve identification of land use types, but it was decided that there was most likely to be an effect over paddy soil types, where other crops would be expected to be totally inappropriate. Thus it was decided to test the hypothesis by adding only the information on paddy soil to the minimum distance classification. This was done by preparing an image mask of the paddy soil area, by contouring all pixel values beyond the paddy soil areas to zero. The paddy soil area itself was set to an appropriate value (20) and the whole image was subtracted from the classified image to produce an image with two additional classes as given below:

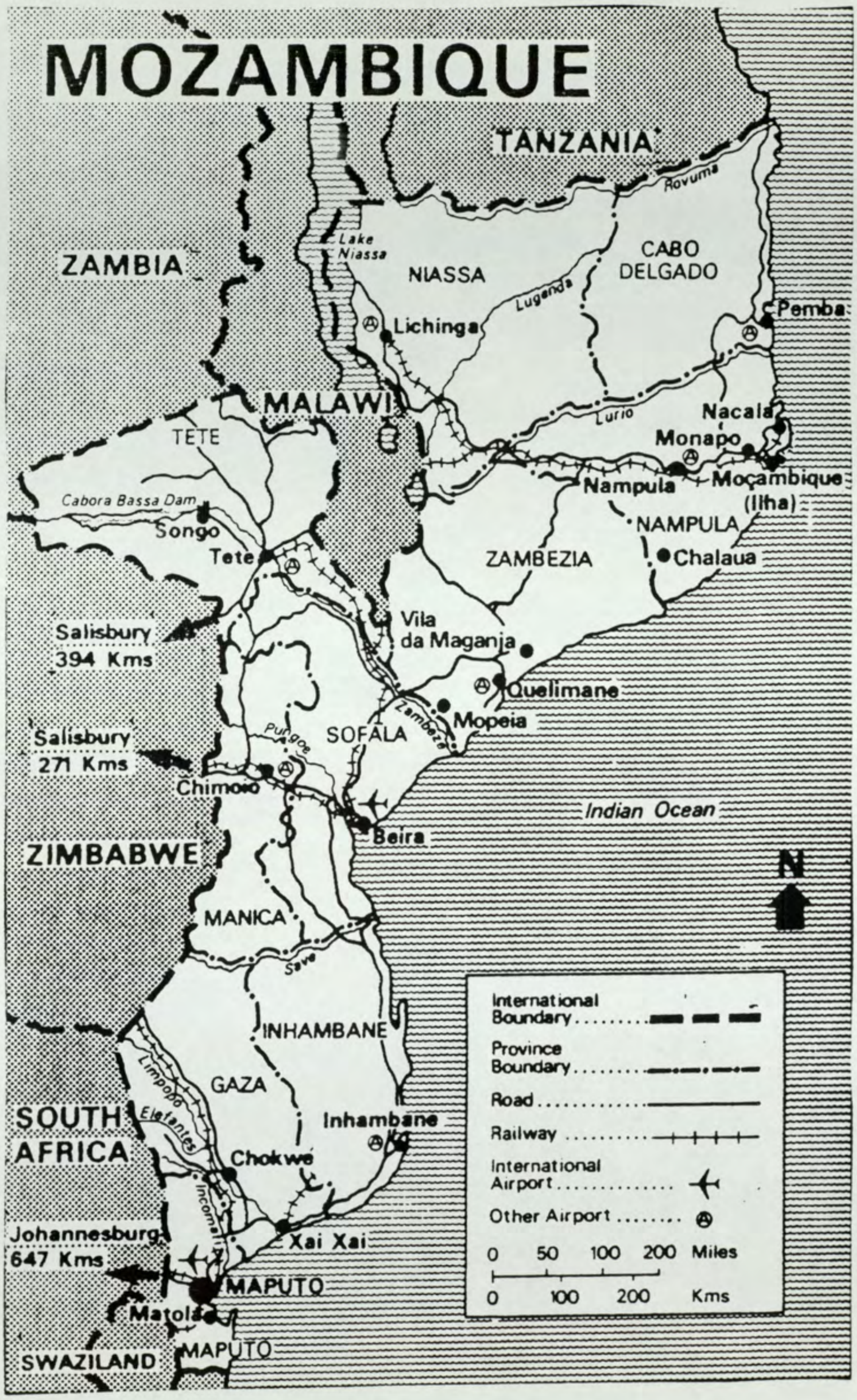
1. Paddy soil but no paddy rice detected.
2. Paddy soil and paddy rice detected.

Paddy rice areas on the original classification, where there was no paddy soil, still remained as a distinct class.

4.8.2.2 <u>Mozambique Case Study</u>

In May 1988, ULG were asked to provide an analysis of satellite image data, for a provincial project in Mozambique. The project concerned was to establish a plan for an integrated rural development programme, for Manica province in the South West of Mozambique. The province covers an area of 61,000 Km^2 along the border with Zimbabwe and is bisected by the Beira corridor as shown in figure 4.7. Originally there had been no intention to use remote sensing but the need arose to confirm the current land use pattern. No mapping of current vegetation at a relevant scale was in existance and there was no air photo coverage. Funds were available for the purchase of

Figure 4.7 Showing the location of the Mozambique study area of Manica province.



satellite imagery, so this provided a perfect opportunity to examine the relevance of satellite remote sensing to a land use planning problem, typical of those encountered in the ULG sphere of work. The main objective was to obtain a preliminary classification of land cover. The intention was to provide support information to aid in stratification of the province, into zones of importance to the strategic plan. The client was also interested in soil conservation aspects related to bare ground and slope of the land, for erodibility assessments. The possibility of developing a GIS for the province, with data on all the natural resources was also mooted.

Fundamental to understanding the utility of satellite imagery, is the fact that what is obtained from a quantitative interpretation of the imagery is a grouping of areas on the ground, with similar spectral reflectance characteristics. The spectral classes resulting from a classification, may or may not correspond to the classes of ground cover of ultimate interest. Often within an area of particular land use, several classes occur resulting overall, in a heterogeneous spectral signature for the land use class. A field visit is required to establish the correlation between spectral classes and land cover. Ground survey mapping of land cover over a small representative area may also improve the classification and allow some measure of accuracy to be given, but for cover classes such as annual crops, a problem usually arises in collecting field data and satellite data simultaneously.

Without ancillary data, processing of satellite data can only provide information on land cover. With local knowledge, or other data sets it may be possible to infer other information from the spectral classification. For example to map erosion risk, the location of bare ground might be determined from the imagery and by combination with other data sets such as soils information, rainfall distribution and slope classification, an erosion map could be prepared.

A strict time schedule of one month for completion of the remote sensing input in Mozambique, limited the analysis to the main objective of providing a map of land cover and the consideration of developing a spatial analysis system as a paper exercise. The security situation in Mozambique with most of Manica province being under rebel influence, meant that the only local work that could be safely conducted was over-flying of the province, to validate the classes obtained by classification of the satellite data.

Standard procedures were followed using processing routines written in FORTRAN. The stages in processing the data are given below:

1. Stripping of non-image data and formatting for processing.

2. Unsupervised clustering to determine natural groupings.

3. Maximum likelihood classification on all four bands of image data.

4. Digitisation of provincial boundary and rectification to each image scene.

5. Masking of classified image by boundary image and calculation of number of pixels in each class to supply area estimates.

6. Geometric correction of classified images to a N-S orientation, using an affine transform.

7. Printing of classified images, after modal filtering.

CHAPTER V

RESULTS

5.1 Micro Expert

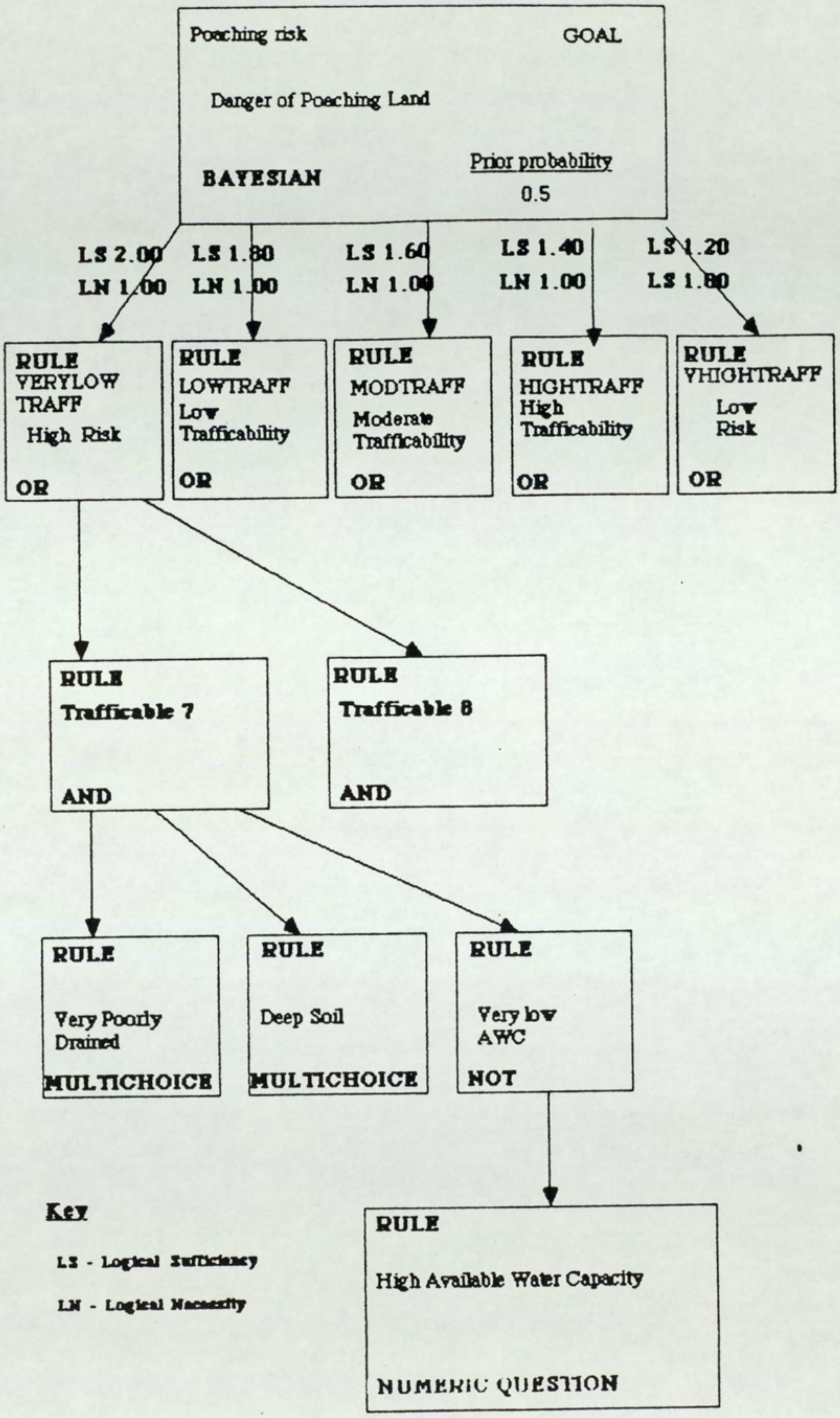
The layout of the land trafficability system designed is shown in figure 5.1 and the full listing is provided in appendix A. The problem chosen for investigation was simple and small enough to be easily envisaged, but it still required a representation compatible with larger problems. The results are presented in terms of the adequacy of knowledge representation.

Micro Expert is an interactive system so the database (situation model), representing the information on the current problem, is supplied by the user through answering questions. As questions are answered, so the probability of the goal hypothesis changes until all pertinent questions have been answered and a final probability for the goal being true is given. Thus the database of information at any point is reflected in the propagation of probability values through the rule network.

All known facts must form a part of the knowledge base made up of production rules in "If - - - THEN" form. There is no clause for simply inputting established facts, except for numeric values which can be assigned as constants. The 'Advice language" was found to be very restrictive in development of a classification system, because of the limited type of constructs allowed.

Crucial to the representation of a classification system, is the facility to input data in the form of a category level for a given variable and the means of relating this to output, also in categorized form. An example would be soil wetness class for which there are six possible

Figure 5.1 Showing the hierarchical structure of the land trafficability problem designed in Micro Expert



categories. Input can be achieved through the use of modulus rules in conjunction with a numeric question, although this does not maintain the ranking, which means further transformations are required to re-establish the meaning of the values. Even for a small classification system as in this case the number of rules required quickly multiplies, because of the limited syntax.

Output in category form such as a choice of suitability class for a selected land use is even more difficult to achieve. It can be done by using a Bayesian rule in a very contrived form. The Bayesian rule is designed to combine factors affecting an outcome, in such a way that the relative importance of each factor is expressed. It can be used in this sense to yield a suitability category, according to the value of the land quality concerned. The value is reflected by the certainty factor assigned to the quality. This can require several complicated dummy rules to transform the input into the required form.

Even the most simple numeric calculations require lengthy programming in the advice language. The basic operators are provided: for addition; subtraction; multiplication and division. 'Less-than' and 'greater-than' can also be achieved through the correct setting of a range rule. Within the advice language it is impracticable to include any but the simplest functions.

An external function call to routines in PASCAL, can in theory be used but it is reported to be clumsy and difficult, it was not available under the system used.

Unlike most expert systems Micro Expert is separated into a rule language compiler called EXPCOMP and a run-time system called RUNEXPT. The requirement of compilation slows down development. This together with the necessity to learn a new language, means that building a system, certainly for the first time, can be a long task. The small system designed here took approximately 7 man days (m/days) to build. This included a substantial `training' stage after syntax errors were corrected. The training was required to `tune' the LS and LN factors, to give the correct answers. Despite having calculated the required LS and LN values, to give the correct certainty range for Poaching risk given particular conditions, the first results were not consistently correct. This training stage was required because the propagation of probability through the tree structure had not been correctly predicted. This in turn is a reflection of the complication introduced by putting complete emphasis on expression of probability. The effects of a particular combination of answers are not immediately transparent to the user.

5.2 **Farmer Interviews**

A summary of the information collected by questionnaire in Rayong province is given in appendix G. For each interview informal notes were also taken, covering responses to particular additional questions which arose in discussion with the farmer. These notes were used in devising the rules for the prolog system listed in appendix E.

Cooperation on interviews was generally good, with only one case encountered where the farmer was definitely trying to avoid answering questions. The use of a pre-set questionnaire meant standard information was collected but this sometimes took a long time. Lengthy interviews were usually caused by a complicated cropping system with many crops involved, or problems of recollection in the case of some of the older farmers. Others had great difficulty in expressing quantities in recognised units such as kilograms. These complications often led to compromises on the information collected by omission of detailed labour patterns for example. Much of the pre-set data was in any case background, for economic calculations, or designed to aid in assessment of a complex factor. For example, the availability of family labour was considered important and this was assessed by detailing the labour pattern for a particular crop as shown in table 2 of the questionnaires (see appendix G).

The most important information for building the computer systems could have been collected in no more than half an hour with each farmer, by informal interview. Unfortunately this would have been difficult to achieve without dispensing with the standard questions. Translation effort was concentrated on explaining the questionnaire, particularly since the farmers often did not understand the meaning of questions. Many farmers had no clear knowledge of their yield from a crop (or were reluctant to be honest about it). Much time was spent in trying to quantify such measures, which were not actually central to the data collection exercise. This left little time to question farmers on the detail of their cropping systems and why they adopted particular practices. In contrast to their muddled calculations of yields, most farmers proved very informed on cropping practice, even if much action was based on traditional concepts.

5.3 Inductive Expert System Shells

5.3.1 Expert Ease

For the first system, developed on the Labi (Brunei) data, attribute values for examples were given as they appeared on the data sheets, with no attempt made to categorise data. The rule induced from these examples is shown in figure 5.2 and the decision tree is represented in more conventional form in figure 5.3

Figure 5.2 Showing the rule induced from Labi examples

```
TEXTURE
      C: SLOPE
          <2 : SLOPE
                 <1 : PADI
                >=1 : HORTICULT
         >=2 : SLOPE
             <5 : FRUITREES
            >=5 : DRAINAGE
                  <7 : DIPTFOREST
                 >=7 : RUBBER
    CL : null
  SiCL : null
    SL : DRAINAGE
          <4 : WATERTAB
               <=70 FRUITREES
         >=4 : SLOPE
                 <7 : DIPTFOREST
                >=7 : RUBBER
 FSL : RUBBER
 CSC : DIPTFOREST
  PY : PEATSWAMP
  MY : PEATSWAMP
 SCL : FRUITREES
  SC : SLOPE
       <29 : SLOPE
              <18 : HORTCULT
             >=18 : RUBBER
      >=29 : DIPTFOREST
 FSL : SLOPE
         <8 : WATERTAB
        <58 : DIPTFOREST
       >=18 : FRUITFOREST
 FSC : SLOPE
       <=8 : RUBBER
 CSL : DIPTFOREST
   L : HORTCULT
 PYC : PEATSWAMP
 MYC : PEATSWAMP
```

<u>Key</u>

TEXTURE (Soil Texture)
C - Clay
CL - Clay loam
SiCL - Silty clay loam
SCL - Sandy clay loam
SL - Sandy loam
FSL - Fine sandy loam
MY - Muck

CSL - Coarse sandy loam
L - Loam
FSC - Fine sandy clay
PYC - Peaty clay
MYC - Mucky clay
SC - Sandy clay
PY - Peat

Depth to water table is in cm and slope is in degrees.
<u>OUTCOME - LANDUSE</u>
DIPTFOREST - Climax Forest
FUITREES - Fruit trees
HORTICULTURE - vegetables

PADI - Wetland rice
PEATSWAMP -Climax peat swamp
RUBBER.

Figure 5.4 Showing the example set and new rule listing with completely categorised data.

```
TEXTURE
      C : SLOPE
             <1 : WATERTAB
                    <4 : PADI
                   >=4 : DRAINAGE
                            <3 : PADI
                           >=3 : HORTCULT
            >=1 : SLOPE
                     <2 : FRUITREES
                    >=2 : DRAINAGE
                            <7 : DIPTFOREST
                           >=7 : RUBBER
     SL : SLOPE
            <3 : DRAINAGE
                  <4 : SLOPE
                        <1 : WATERTAB
                            <4 : FRUITREES
                           >=4 : WATERTAB
                                  <5   : HORTCULT
                                  >=5 : FRUITREES
                        >1 : HORTCULT
                 >=4 : WATERTAB
                      <4  : RUBBER
                      >=4 : SLOPE
                           <1 : CLASH(FRUITREES, RUBBER)
                           >=1 : FRUITREES
            >=3 : DRAINAGE
                   <7   : DIPTFOREST
                   >=7 : RUBBER
     PY : PEATSWAMP
      L : SLOPE
              <4 : SLOPE
                     <2 : WATERTAB
                             <3  : DIPTFOREST
                             >=3 : FRUITREES
                    >=2 : HORTCULT
                 >=4 : SLOPE
                        <5  : RUBBER
                        >=5 : DRAINAGE
                           <6  : CLASH (RUBBER DIPTFOREST)
                           >=6 : CLASH (RUBBER DIPTFOREST)
```

Categorisation of depth to water table and slope

WATERTAB		SLOPE	
Class	Depth	Class	Gradient
0	0-19cm	0	0-1°
1	20-39cm	1	2-4°
2	40-59cm	2	5-6°
3	60-74cm	3	7-11°
4	75-99cm	4	12-24°
5	100-149cm	5	=>25°
6	=>150cm		

Table 5.1 shows which limits were actually considered important by experts, in determining crop requirements for land utilisation types, in the Labi area. Comparison with figure 5.2 show there is some broad agreement, although the decision tree is full of inconsistencies, because of the small sample set. Also, some of the deficiencies are caused by the effects of outliers, the unrepresentative examples. Increasing sample size will not affect treatment of outliers, they will always have an exaggerated effect, causing class limits to be broadened to encompass them and thus leading to problems in differentiation, due to increased overlap of classes for different crops.

Inconsistencies may not necessarily be a problem if the system is used by an expert. Possible limits would still be identified and the expert can further improve the classification, by using those limits to define new categories for attributes.

This categorisation task was done for the labi data and a new rule was induced from the adjusted set of examples. The categorised example set and new rule listing are shown in figure 5.4.

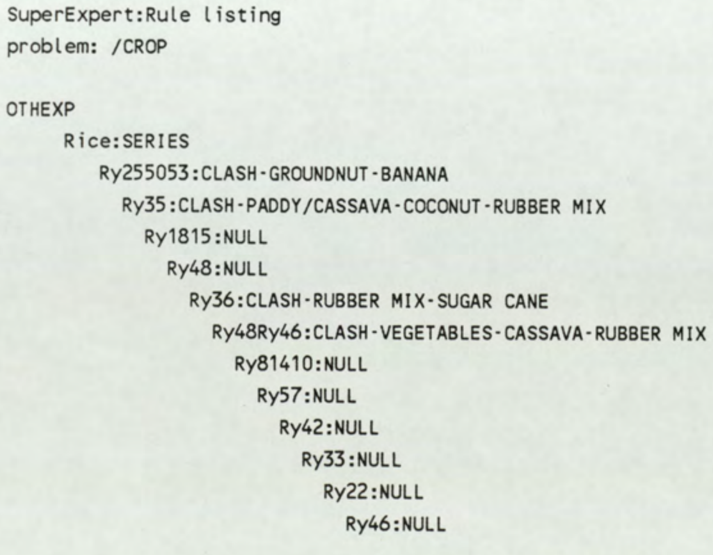
5.3.2 Super Expert

5.3.2.1 <u>Rayong Province</u>

The attributes available from the Rayong questionnaire are listed in appendix B. In contrast to the yasothon data the land units for Rayong had not yet been decided at the time of the survey, but the soil series was identified for each site and in the DLD this is used as the chief factor in setting land units (see appendix I). The drainage and physiography as well as texture can be inferred from the series. An independent measure of slope was made at each site. An important agroclimatic division also crosses Rayong province approximately N-S, see figure 5.6. The rainy season is prolonged to the East of the boundary (5-7 months compared to no more than 5 in the west) and rains are often of heavier intensity. The Eastern part of the province was designated `MOIST' when detailing examples. These three factors, labelled: `MOIST'; `SERIES'; and `SLOPE' together with `WATER' (depth to water table) should have been sufficient to summarise the physical status of the land.

The first example set tried (listed as Rayong 1 in appendix B) was actually of 60 examples and soil series was included as such, giving the specific soil code for a site. This gave a rule of the form shown in figure 5.5 and listed in full in appendix B as rule Rayong 1. Note the domination of the soil series factor as the second attribute to be considered. It can be seen from the form of the rule, that no complicated hierarchy has been detected, only two attributes are found to be important in choosing the crop for most cases. The other attribute used once is `AGE'.

Figure 5.5 Screen dump from part of the Rayong 1 rule.

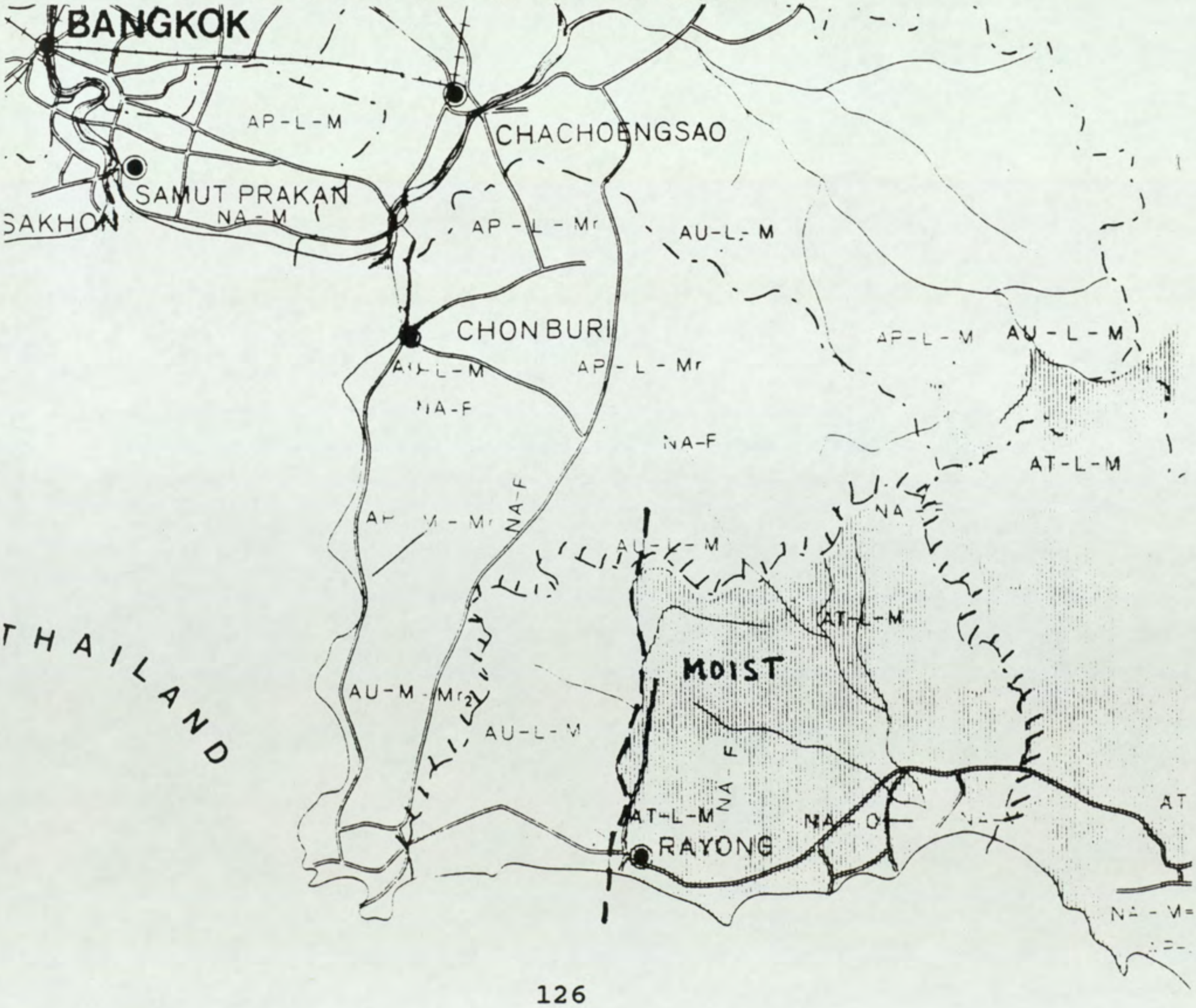


```
SuperExpert:Rule listing
problem: /CROP

OTHEXP
     Rice:SERIES
        Ry255053:CLASH-GROUNDNUT-BANANA
          Ry35:CLASH-PADDY/CASSAVA-COCONUT-RUBBER MIX
            Ry1815:NULL
              Ry48:NULL
                Ry36:CLASH-RUBBER MIX-SUGAR CANE
                  Ry48Ry46:CLASH-VEGETABLES-CASSAVA-RUBBER MIX
                    Ry81410:NULL
                      Ry57:NULL
                        Ry42:NULL
                          Ry33:NULL
                            Ry22:NULL
                              Ry46:NULL
```

The large number of value categories for the soil series was suspected to be having an effect on the rule formed, so the series were grouped into only five classes based on physiographic position and parent material. The crops for which only one example was available were also removed from the listing to leave only 55 examples (given as Rayong 2 in appendix B).

The number of possible combinations of attribute values, considering logical attributes alone, is 1,290,240 (assuming complete independence). The 55 examples available, on this calculation, thus represents less than 0.004% of the complete domain of instances.

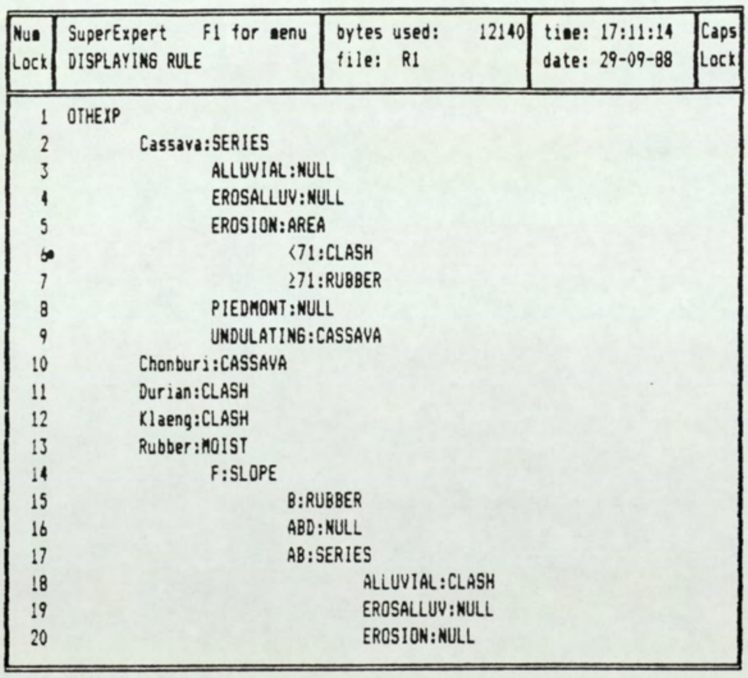
Figure 5.6 Showing the agroclimatic conditions in Rayong province (after DLD, 1981).



The first portion of the rule formed from the Rayong 2 set, is shown in a printout of a screen dump in figure 5.7. The complete rule is given in appendix B. The rule took three minutes to induce and had 51 nodes (decision points). There were 16 clashes in outcome and 13 null nodes.

The second rule listing with categorisation, shows a more hierarchical decision tree. It is interesting that the attribute of 'OTHEXP' representing previous experience is found to be the primary factor in determining crop choice in both cases. This was not expected and was not immediately obvious from the example listing.

Figure 5.7 Screen dump from part of the Rayong 2 rule.



```
Num  SuperExpert    F1 for menu   bytes used:    12140   time: 17:11:14   Caps
Lock DISPLAYING RULE              file:  R1              date: 29-09-88   Lock

  1  OTHEXP
  2          Cassava:SERIES
  3                  ALLUVIAL:NULL
  4                  EROSALLUV:NULL
  5                  EROSION:AREA
  6                          <71:CLASH
  7                          ≥71:RUBBER
  8                  PIEDMONT:NULL
  9                  UNDULATING:CASSAVA
 10          Chonburi:CASSAVA
 11          Durian:CLASH
 12          Klaeng:CLASH
 13          Rubber:MOIST
 14                  F:SLOPE
 15                          B:RUBBER
 16                          ABD:NULL
 17                          AB:SERIES
 18                                  ALLUVIAL:CLASH
 19                                  EROSALLUV:NULL
 20                                  EROSION:NULL
```

5.3.2.2 Yasothon Province

Preparation of the data from database information took approximately four man days. The original data was collected over the space of 120 m/days.

For Yasothon, with 289 examples extracted from a complete data set of 617 records, an induction rule was formed after six minutes with 158 nodes (decision points), 16 clashes in outcomes and four null nodes (no outcome defined). An excerpt of the rule with explanation is given in figure 5.8, the full rule is given in appendix B. The data actually contained some duplicate examples that could not easily be removed from the database used. This error was introduced in the combination of data from four separate databases to form the example set. No unique questionnaire number was available, so combination of the data led to difficulty in removing the duplicate examples (by the otherwise simple application, of the DBASE 'UNIQUE' command on the key field). Questionnaire numbers were only unique to the particular land unit. Remedying this problem could have been time consuming, but an advantage of Super Expert is that it can handle such circumstances, by checking against preceding examples as new examples are added. This gives the operator an opportunity to eliminate duplicates. Alternatively, for speed, the duplicates can be included automatically with a slight increase in the time required for the induction process. The latter option was chosen. The number of unique examples was 188.

For the logical attributes chosen, the number of possible factor-value combinations was 144. The splitting of integer attributes into a number of classes is not easily predicted and will be different for each branch of the decision tree (see explanation of the algorithm in section 3.2.7 and appendix B), which makes calculation of the total domain of possible combinations imprecise.

Figure 5.8 Showing an excerpt from the rule formed by induction on the Yasothon data.

```
LU
  Y5 : WATER
     RAINFED : TENRAII
          < 4 : JOBNO
            < 2 : AGE
              < 46 : MALIMEL
              >=46 : DOYRS
                 < 33 : GLUT
                 >=33 : MALIMEL
           >= 2 : MALIGRD
         >= 4 : DOYRS
           < 19 : MALIMEL
           >=19 : TENRAII
              < 15 : AGE
                 < 39 : MALICHICOR
                 >=39 : AGE
                    < 42 : MALIGRD
                    >=42 : DOYRS
                       < 36 : MALICHICOR
                       >=36 : MALIGRD
                >= 15 : AGE
                    < 51 : MALIMEL
                    >=51 : MALICHICOR
```

Key

Attributes

LU	-	Land Unit
WATER	-	Water regime: rainfed, irrigated, or both.
JOBNO	-	Number of other jobs held by the family.
AGE	-	Age of the farmer.
DOYRS	-	Number of years spent farming.
TENRAI1	-	Tenure status on the major farm plot.

Crops

MALIMEL	-	Paddy rice (Mali variety) rotated with melon in the dry season.
GLUT	-	Glutinous rice.
MALIGRD	-	Paddy rice, rotated with groundnuts.
MALICHICOR	-	Paddy rice-Chilli-Corn, grown in rotation.

Assuming only 4 categories for the three most important integer attributes, this would lead to a possible 9,216 different instances. In which case the example set used would represent less than 2% of the possible situations. From inspection of the rule formed, the main integer attributes used were actually divided into more than four value categories.

The four null nodes, where no outcome could be predicted, are listed below :

1 'Land Unit (LU)=Y5' + 'Water Regime (WATER)=IRR+RAIN'
  ---> NULL

2 'Land Unit (LU)=Y6' ---> NULL

3 'Land Unit (LU)=Y11'+'PLOT SIZE (TENRAI1) >=9 and <18'+
  'Experience (DOYRS) >=5 yrs.' + 'EDUCATION=A'---> NULL

4 'Land Unit (LU)=Y13' ---> NULL

Nodes 2 and 4 occur because no examples were included in the data set for land units Y6 and Y13. There were occurrences of these land units in the complete data set and this illustrates one of the problems of deciding on the training set as discussed in section 7.3.2.2 of the discussion. Node 1 occurs because there is no example of that particular water regime for land unit Y5. Neither is there any instance in the complete data set of 617 records. This might be taken to indicate that this combination never occurs but since irrigation and rainfed systems are both practiced on this land unit, a combination of the two would seem possible. For the purposes of this expert system the category of 'IRR+RAIN', meaning a combination of rainfed and irrigation (probably only supplementary irrigation), may not be an appropriate logical value. This is then an example of how at a later stage the system could be improved, by removing this distinction from the training set. Node 3 illustrates the case where more examples must be sought, to identify whether education level is likely to effect the choice of crop. For the crops considered, it would seem unlikely that the level of education would have a direct effect on the ability to grow one of the possible crops. An effect seems to have been detected, but there were only a few people in the sample with an education level of class 'A'.

The integer attribute JOBNO (number of other jobs held by the family) was only used once and the logical attribute TEN1 (tenure status) was never used to form the rule. The other attributes were used in several branches of the decision tree. The significance of this and the ordering of attribute importance is discussed in section 7.3.2.2.

5.4 TIMM

The rule listing for the decision support system written using TIMM is given in appendix C. The rule for the Rayong crop choice system, was derived from analysis of questionnaires from the second field visit made. The system was tested against the questionnaires collected on the last field visit. The results are shown in table C.1 of appendix C.

Like the induction systems, the aim of using TIMM was to arrive automatically at a system, to accurately emulate the way farmers in Rayong province would choose a crop. The results can be compared directly with the induction systems in that the same example set was used, as that in the Super Expert Rayong system.

The approach contrasts with that adopted for the prolog system for Rayong, where the rules were decided on the basis of all the information available from a set of just over twenty interviews, rather than only from a pre-set answer format. For the TIMM system informal interview information was only used in the identification of attributes.

In principle it should be possible to build a system very quickly in TIMM, but in practice the very primitive editor in the software causes problems. Corrections are not easily made, necessitating in some cases the re-entry of attribute information. Despite this shortcoming the design and development of the Rayong system in TIMM only took about 5 m/days, of which two days were spent on designing and re-designing the structure of the system. At first a deeply nested hierarchical design was favoured, but this would have entailed duplication of questions and so was abandoned.

It proved relatively easy to build the system in TIMM, no programming was involved. For the 'farmer decision' system the important attributes were decided on the basis of knowledge gained interviewing farmers, i.e. personal experience. Having decided on the attributes to be evaluated, values of those attributes were entered as examples from the questionnaires, together with the crop grown. Where a farmer had different plots of land, an entry was often made for each plot, especially if one plot was very different, planted to paddy rice for instance. Otherwise the percentage of the holding down to a particular crop was appended as a certainty factor for that crop. Because of difficulties in collecting complete information for every farm plot (as described in section 5.2), some entries were not accurate. For example spatial variation in depth to water table and slope, was not often recorded and labour profiles were sometimes omitted. In such cases the information as it appeared on the completed questionnaire was entered even though there was doubt, as to say the true slope on a particular plot. This kind of problem is often encountered in processing interviews, so it was instructive to see how such inaccuracies would affect the system.

A similar problem occurred on exercising the system, with other completed questionnaires not used in the training, information was sometimes not accurately recorded so that a '?' had to be given in answer to some questions, meaning value unknown. The results of the exercise session are given in table C.1 in appendix C, which indicates how accurately the system actually predicted farmer's decisions. Ten out of twenty nine examples checked, were judged to be accurate predictions. The attributes of distance to the sugar cane factory and distance to the pineapple factory were not actually used, despite their identification as important factors from experience. This was because although farm location was noted, information on factory locations was not available within DLD and could not be readily obtained from the Department of Industry. No doubt the information would have eventually been found somewhere, but this was not achieved in time. This problem had not been anticipated and is an apt illustration of one of the advantages of research into LIS, the identification of information vacuums is very important. Planning LIS top down, such information requirements can go unnoticed. There is the possibility that a complete LIS can be designed and implemented only to fail because the important factors were not identified. From discussions with the farmers, it was evident that the distance to processing factories was very important to their decision, as to whether or not to grow sugar cane or pineapples. The factor was complicated by the possibility of transport provision by the factory and other personal factors, such as ownership of a lorry, but it shouldn't be ignored. Unfortunately for the purposes of the TIMM system and the induction systems, it had to be ignored.

The original intention had been to use TIMM both to apply relevant farmer rules and then as a system for implementation of an FAO style evaluation procedure. TIMM was found to be unsuitable for this latter role for several reasons:

1/ The poor editor complicated the process of building a large system.

2/ An expert system defined in TIMM is limited to 15 choices of outcome. This is not adequate to cover most situations as cropping options usually exceed this number.

3/ A maximum of twenty factors are allowed. This is also not adequate for most agricultural circumstances, although linking of systems can obviate this problem.

4/ It did not prove possible to 'embed' a TIMM system within a FORTRAN application program, so that a TIMM system such as the 'Farmer's choice' could not be linked to an application for more complicated financial analysis.

5/ The above problem of linkage to other software meant introduction of new example rules was a laborious task.

5.5 dBASE

5.5.1 Suitability Evaluation

As was explained in section 1.2, land evaluation following FAO procedures is essentially a matching process between crop requirements and the qualities of land units. The overall qualitative suitability rating is determined by the most limiting factor and all factors must be considered together, they are treated as independent.

To arrive at a suitability rating in a land evaluation exercise, requires the conjunction of values for a number of relevant land quality values, by use of the AND operator. By Boolean logic the lowest value will determine the final rating, which is consistent with normal practice of judging the land on the most limiting factor. NOT; GREATER THAN and LESS THAN are often required for comparing land quality values against a threshold defined by the land use planner.

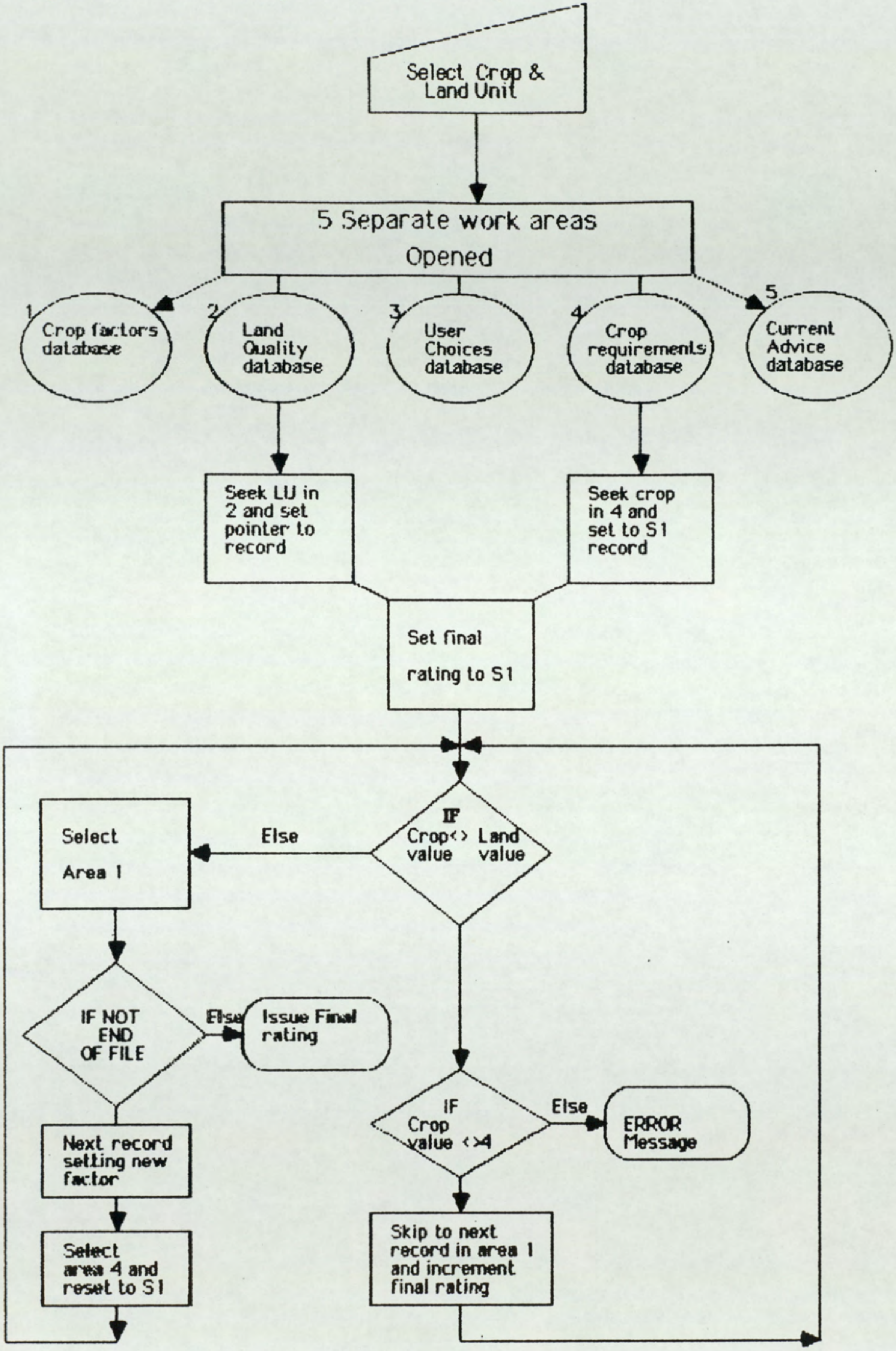
The algorithm for the DBASE III program simply considered each crop factor in turn, assigned a rating and compared the rating to that of the next factor, if the next factor had a lower rating that was stored as the overall rating, if not the previous rating value was used. The flow diagram in figure 5.9 shows the logical path. The crop requirement information was derived from DLD guidelines and standard texts such as De Geus (1973).

Once derived, the final qualitative suitability rating was available to the spreadsheet macro, written in LOTUS 123 to accomplish a calculation of net present value (NPV). A menu program written in DBASE III (which actually required a lengthier program than the evaluation program) gave the user options to select the economic evaluation.

5.5.2 Program Description

A modular style was adopted for the programming task, in that the problem was split into each of it's component parts as shown in the flow diagram in figure 5.9. Each component was then coded as a procedure in the procedures file CHANGES.PRG. The modules could have been coded as individual programs but this would have led to more intensive disk access during program execution. By setting the procedure to CHANGES in the EVALUATE program, all the subroutines (procedures) in the CHANGES file were loaded into memory at the beginning of the EVALUATE program and run from memory, which is a faster method. The full listing of the program files and structure of the database files, used in the DBASE Land Suitability Evaluation system are given in appendix D.

Fig. 5.9 Flow diagram of DBASE system


Select Crop & Land Unit
5 Separate work areas Opened
1
Crop factors database
2
Land Quality database
3
User Choices database
4
Crop requirements database
5
Current Advice database
Seek LU in 2 and set pointer to record
Seek crop in 4 and set to S1 record
Set final rating to S1
IF Crop<> Land value value
Else
Select Area 1
IF NOT END OF FILE
Else
Issue Final rating
Next record setting new factor
Select area 4 and reset to S1
IF Crop value <>4
Else
ERROR Message
Skip to next record in area 1 and increment final rating

5.5.2.1 The 'Front End'

The crop and land unit selected by the user, to obtain recommendations, are stored in a database called SELECTIO.DBF. This allows several different queries to be made at once and queued in the database. The entry screen displayed to the user is shown in figure 5.10.

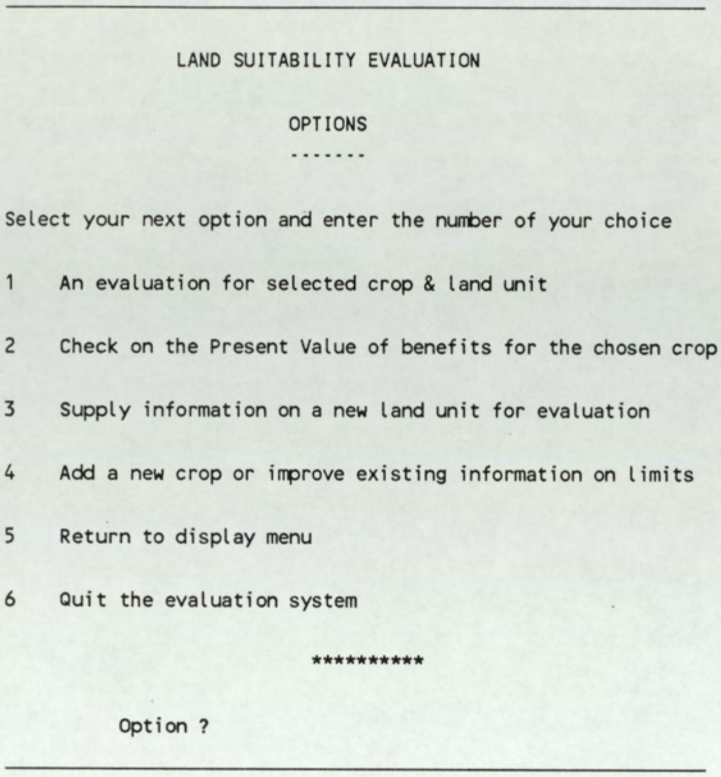
The options screen is provided to allow selection of other options, associated with the main land evaluation assessment. The options screen also allows the knowledge base of the evaluation system to be extended by adding new land unit or crop information.

For the 'OPTIONS' menu the user was asked to enter a number which was stored to a memory variable. This was achieved using the construct shown below:

```
@[Position] SAY [Message] GET [Variable] PICT [Format]
```

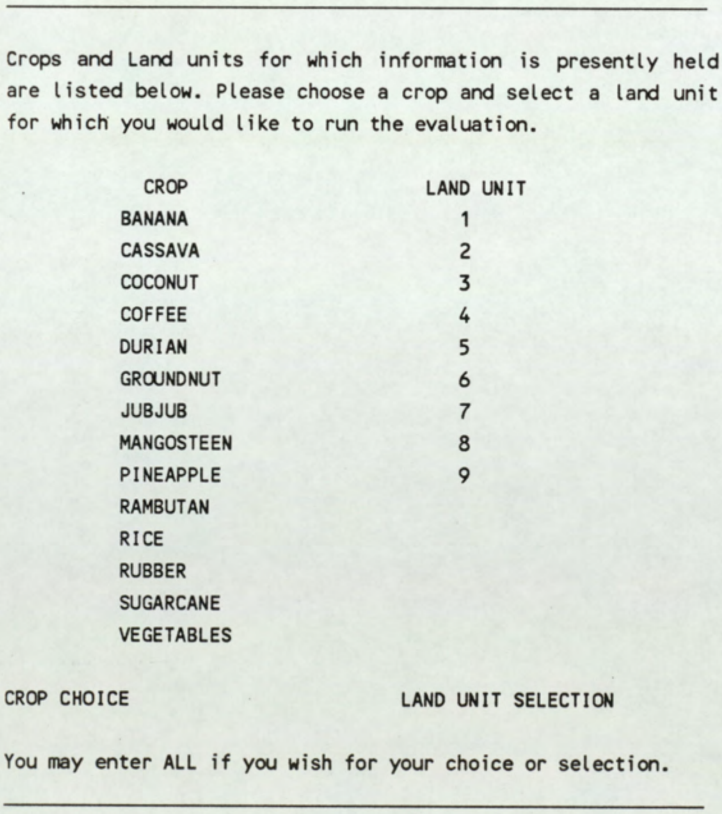
The interpretation of these commands is: At a specific screen position display the message to the user and store his reply to the variable, provided the reply meets the format constraints. In the case of the options screen, the format constraints were that the reply should be a number, in the range 1-6.

Figure 5.10 Screen dump of the 'Options' screen



```
                LAND SUITABILITY EVALUATION

                           OPTIONS
                           -------

Select your next option and enter the number of your choice

1    An evaluation for selected crop & land unit

2    Check on the Present Value of benefits for the chosen crop

3    Supply information on a new land unit for evaluation

4    Add a new crop or improve existing information on limits

5    Return to display menu

6    Quit the evaluation system

                             **********

         Option ?
```

Figure 5.11 Screen dump of the 'Selection' screen.



```
Crops and Land units for which information is presently held
are listed below. Please choose a crop and select a land unit
for which you would like to run the evaluation.

           CROP                      LAND UNIT
         BANANA                          1
         CASSAVA                         2
         COCONUT                         3
         COFFEE                          4
         DURIAN                          5
         GROUNDNUT                       6
         JUBJUB                          7
         MANGOSTEEN                      8
         PINEAPPLE                       9
         RAMBUTAN
         RICE
         RUBBER
         SUGARCANE
         VEGETABLES

CROP CHOICE                        LAND UNIT SELECTION

You may enter ALL if you wish for your choice or selection.
```

The number stored in the memory variable was then checked using a 'CASE' construct. 'DO CASE', is a DBASE command used to follow one path of actions from a choice of several options. Each option is a 'CASE', which will only be instigated if a certain variable value is met, only one 'CASE' can proceed. The code excerpt given in figure 5.12 shows how the '@- -SAY- -GET- -PICT' and 'DO CASE' constructs were used to program the Selection of options. For example if option 1 was entered by the user, this would trigger 'CASE O=1'. In that event databases in two different work areas would be cleared ready for new information and two procedures; REQUEST and STANDARD, would be instigated successively.

The 'REQUEST' procedure also makes use of memory variables to store replies, but only to guide which action to take next. The selections of crop and land unit made from the selection screen (see figure 5.11), are stored directly to the 'SELECTIO' DBASE file, again using an '@-SAY-GET-PICT' construct, but with field names instead of memory variables. To allow several selections to be made the 'GET' clauses were included in a 'DO WHILE' loop, allowing command statements within the loop to be repeated, provided a specified condition is true, the condition was a check on a memory variable. The memory variable value was dependent on the users option to make another selection. This was programmed with a WAIT command. The WAIT command which displays a prompt, pauses output until a key is pressed and the key value can be stored to an associated variable (see PROCEDURE REQUEST in appendix D).

Figure 5.12 Showing code from OPTIONS PROCEDURE

```
@ 22,20 SAY "  Option ? " GET O PICT "@Z ##" RANGE 1,6
READ          *Read the chosen user option
CLEAR         *according to the format constraints
DO CASE
*     According  to the option chosen, open required databases and
select  relevant  work areas, then 'DO' the appropriate procedures.
Only one of the case options will be triggered.
*
     CASE O=1
          SELE 5
          USE CURRSUIT INDEX CURS
          ZAP
          SELE 3
          USE SELECTION ALIAS SEL
          ZAP
          DO REQUEST
          DO STANDARD
     CASE O=2
          DO CHECKPV
          DO OUTOLOT
          DO PVDISPLAY
          DO OPTIONS
     CASE O=3
          DO USERASK
          DO STANDARD
     CASE O=4
          DO ALTREQ
          DO OPTIONS
     CASE O=5
          DO DISPLAY
     CASE O=6
          SELE 5
          ZAP
          SELE 3
          ZAP
          CLEAR ALL
ENDCASE
```

5.5.2.2 Work Areas

Once the user has entered his problem selection the rest of the procedures revolve around the task of matching land qualities of the selected land unit, to crop requirements of the chosen crop. To achieve this several databases have to be compared and the results stored in another database. This requires that a number of databases should be accessible at once, so five separate work areas were opened.

Table 5.2 shows the databases used and their purpose. The database structures are summarised in appendix D.

Table 5.2 Showing the databases used in the land evaluation system by the EVALUATE program.

Name	Alias	Purpose
LANDINF.DBF	LAND	Stores information on land units.
CROPSUIT.DBF	REQ	Store information on crop requirements.
FACTORS.DBF	FACT	Lists land qualities involved.
SELECTIO.DBF	SEL	Stores users choice of crops and land units for investigation
CURRSUIT.DBF	CURR	Stores current rating for each crop and land unit.

For ease of identification of field names from different databases, 'ALIAS' names were given to the databases. After problem selection the 'STANDARD' procedure was followed. This procedure continued until all queries stored in the 'SELECTIO' database were completed, by use of a DO WHILE .NOT. EOF() construct, which repeats the procedure until the end of the file is reached. The 'STANDARD' procedure directs processing to other procedures, depending on the choices to be investigated i.e. all crops for a particular land unit or a particular crop for a particular land unit. Whatever the choice, the 'PROCEDURE MATCHER' is used at some stage, to match land qualities to requirements of the chosen crop.

5.5.2.3 Matching

The crop requirement and land unit databases share common field names, for the important land qualities to be investigated. Fields of the same name in the two databases, have to be compared for the matching process. The land quality considered at a particular time, was determined by the position of the pointer in the 'FACTORS.DBF', which stored the names of important land qualities. A 'public' memory variable called 'FIELD' was used to store the current quality under consideration.

Declaring a memory variable as 'public' makes it global, so that it can be used and altered by subroutines, without having to pass parameters each time (which was another method used for some subroutines).

Figure 5.13 Showing the code for the MATCHER procedure.

```
PROCEDURE MATCHER *To match land qualities to requirements of the chosen crop.
FINAL=.F.      *Initially set the 'FINAL' variable to False
CRATING=1      *and the rating (suitability) to 1. Thus the initial
SELE 4         *assumption is that the land is very suitable.
DO  WHILE .NOT. FINAL          *Check that the value for the current factor is met
     IF AT(TRIM(LAND->&FIELD),REQ->&FIELD)<>0     *by the land unit considered.
          DO NEWFIELD WITH FINAL,CRATING          *If the value was met check the
     ELSE                                         *next factor, otherwise:
          IF TRIM(RATING)<>'4'                    *If the rating is >4
          SKIP                                    *check against a lower rating
               IF CRATING<VAL(REQ->RATING)        *and adjust the suitability to the
               CRATING=VAL(REQ->RATING)           *lowest so far diagnosed.
               ENDIF
          ELSE
          ?'ERROR, no match'
          CANCEL
          ENDIF
     ENDIF
ENDDO
RETURN
**
```

At the start of the 'MATCHER' procedure shown in the code excerpt in fig 5.13 the suitability rating was assumed to be 1 (CRATING). The 'AT' function was then used for a substring search, to check whether the land unit value for the current quality, was given in the same field in the crop requirements database (**CROPSUIT.DBF**). If there was no match, 'SKIP' was used to move the pointer to the next crop record which would be for an S2 rating and the substring search with the 'AT' command would be tried again. For each crop four records were stored representing S1; S2; S3; and NS ratings with different quality values for each rating, as shown in fig 5.14. The syntax of the 'AT' expression in shown below:

```
AT(<expression 1>, < expression 2>)
```

With the 'AT' substring search, a number is returned which shows the starting position of a character expression within a second expression. If the first substring is not contained within the second character string, zero is returned.

Figure 5.14 Showing how crop requirement records were stored in the CROPSUIT database.

RECORD	CROP	RATING	DRAIN	SLOPE	SURFTEXT	SUBSTEXT	DEPTH
1	COFFEE	1	4	ABC	3	1234567	d5
2	COFFEE	2	35	D	26	1234567	d4
3	COFFEE	3	2	E	26	1234567	d3
4	COFFEE	4	15	F	1	1234567	d2d1
5	RUBBER	1	4	ABC	234	234	d5
6	RUBBER	2	35	D	16	16	d4
7	RUBBER	3	2	E	6	6	d3
8	RUBBER	4	15	F	167	167	d2
9	COCONUT	1	34	ABC	23456	23456	d5
10	COCONUT	2	25	D	16	16	d4

As soon as a match was found the current suitability rating was set to the value of the field 'RATING' in CROPSUIT.DBF and the procedure 'NEWFIELD' was used to select the next land quality for appraisal. For each new land quality, the pointer of the CROPSUIT.DBF was reset to RATING=1 for the chosen crop, by the 'FINDER' procedure using the 'SEEK' command.

5.5.2.4 Display of Results

Display of the results is accomplished through the procedures 'DISPLAY' and 'RESULTS' or 'RSLTSSPRN'. The '@ <Position> SAY- -' command was used to provide explanatory text and options were again selected with a 'WAIT - - To' command, followed by 'DO CASE' construct. The work area was swapped to that of the database for current suitability evaluations (CURRSUIT.DBF), thus enabling the crop, land unit and rating records to be displayed in turn. After display of all the results it was possible to return to the options menu or quit the system. The loop back to the options menu for a new evaluation, or a financial analysis of the suitability evaluation, was accomplished with a simple procedure call to the 'OPTIONS' procedure again, to give the opening menu shown in figure 5.10.

5.5.2.5 Further Options

There was no restriction to further options which could be added to the DBASE evaluation system through more procedures. Although eventually some subsidiary routines would have to be written as independent programs, since in DBASE III a single program is limited to 32 procedures. Only one option was programmed as an example of the method. The option chosen was for the calculation of financial benefit from a crop on a given land unit, by a calculation of net present value (NPV, also known as net present worth). Net present value was used, as this allows for comparison between tree crops and annual crops, by averaging of the present value from tree crops. With so many plantation crops in Rayong such an approach was essential. A net present value calculation allows future cost and benefit streams to be discounted to present values. The net present value is the present worth of the incremental net benefit. It can be calculated, by subtracting the total discounted present worth of the cost stream, from that of the benefit stream, or by discounting the incremental net benefit stream. The reason for using present worth calculations in project analysis, is to take account of the timing of a benefit stream. Other measures commonly used with present worth figures are the benefit-cost ratio and the internal rate of return (IRR). The net present value and benefit-cost ratio measures, will vary with the discount factor (interest rate) chosen. In theory this should be equivalent to the opportunity cost of capital for the country concerned, i.e. that rate which will just result in all the capital in the economy being invested, if all possible projects were undertaken which yielded that much or more return (Gittinger, 1972). In practice the rate tends to be set by 'rule of thumb' and is generally accepted to be between 8-15% for developing countries.

The calculation of net present value implemented, was primarily designed to show the inherent advantage of a spreadsheet, for such financial or economic calculations. Implementation could have been achieved within the DBMS, but there would be disadvantages: calculations would be slower; the method would be less visible to the user; sensitivity analysis by alteration of key values would be less easily performed. The present value calculations are easily understood in a tabular format. Such a format can be maintained in a spreadsheet. In a DBMS the same calculations would have to be implemented in loops, within a program written for the purpose. Because relative positions can be used in spreadsheet calculations, only the first iteration of a repetitive series need be made. The remaining iterations (representing years in this case), are calculated by copying the formula to the rest of the range (see explanation of macro in appendix D for an example of such calculations).

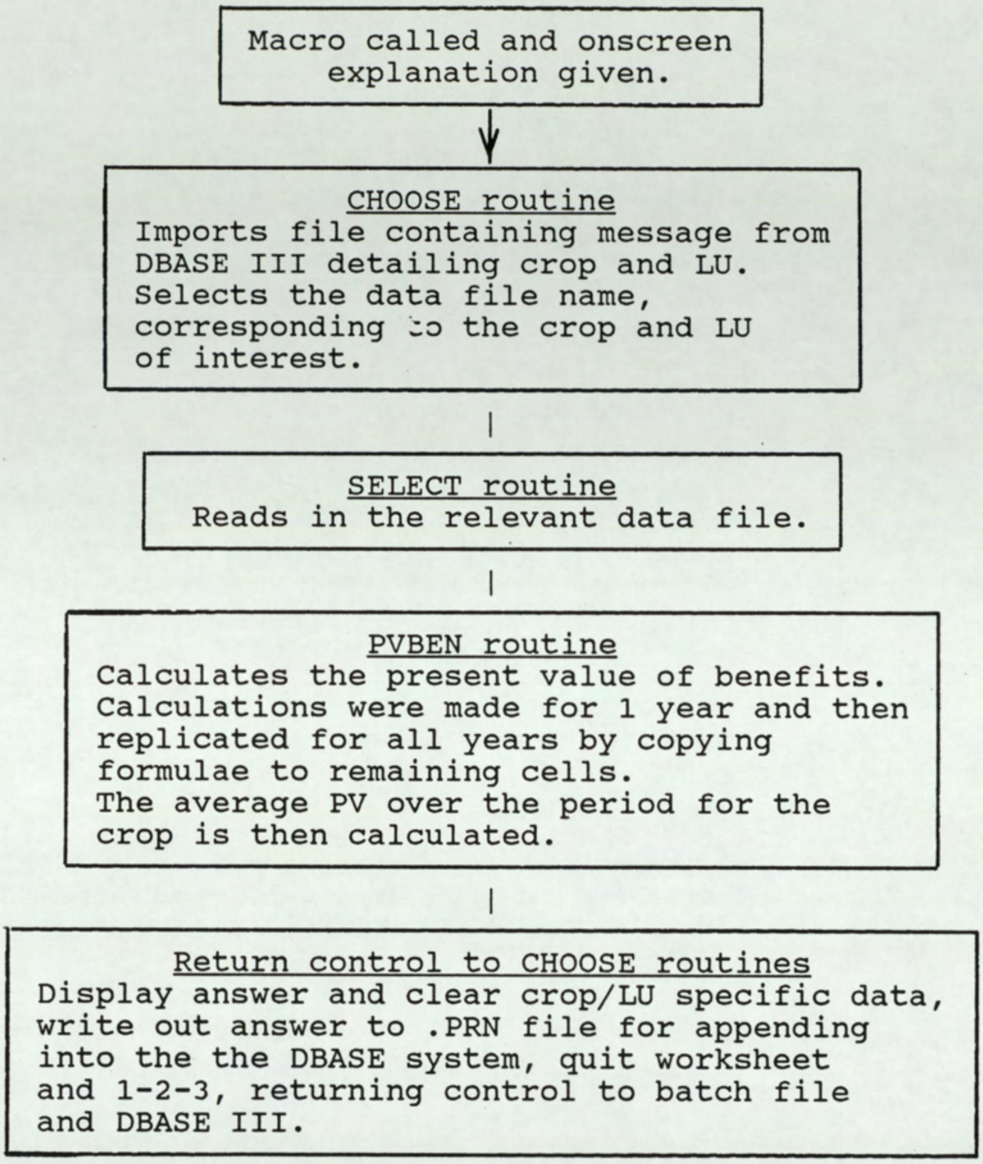
Another option could have been to call a special routine programmed, and perhaps compiled, in a different language. This would still have depended on export of some kind of data file from the DBMS, through DOS and the import of a file with the answer. Since spreadsheets were considered of particular relevance to part of the land use planning problem, the investigation of a link with Lotus 123 was considered the most worthwhile option to implement. As well as illustrating the advantages and disadvantages of such a method, implementation highlighted the problems of making detailed financial analysis on short land use planning projects, where sufficient information is often not available. Despite collection of financial information on prices and costs by the questionnaire for Rayong, sufficiently reliable and complete figures could not be assembled for most crops. Yield figures obtained from farmers showed wide variation and independent confirmation from DLD sources proved difficult for most crops. This is not an uncommon situation to encounter in developing countries. It illustrates the problem of reliance on sophisticated financial or economic indicators, which will have usually been based on some figures which are merely a best guess. Although the NPV calculation was successfully implemented in the macro, none of the actual answers obtained were regarded as very realistic. This in turn indicates how essential the incorporation of an option to perform a sensitivity analysis is, to any automated economic analysis system. This is easily done in a spreadsheet in a manner clearly visible to the user. If economic or financial calculations were to be implemented by another means, the results should be presented in a form allowing and suggesting the application of a sensitivity analysis.

The financial analysis was offered as a selection from the options menu. On selection of option 2, the procedures 'CHECKPV', 'OUTOLOT' and 'PVDISPLAY' were invoked. The 'CHECKPV' procedure opened a sixth work area for the database LOTMES.DBF, which was designed to accept a message from the LOTUS spreadsheet. The message included the name of the current crop under evaluation and the land unit concerned. These details were prepared in the required format for the existing LOTUS field and appended to the LOTUS macro command '/fcce'. This command abbreviation stands for 'file combine copy entire' meaning, copy an entire worksheet into the current worksheet at the current cursor position. The worksheet names were specific to a given crop and land unit, for example DURIAN 1. WK1: The files contained specific prices and costs for that crop on the particular land unit. This message from DBASE was imported into a spreadsheet in LOTUS 123, at the correct position to form a line in a sequence of macro commands (see appendix D for an explanation of macro commands). The macro ran automatically on entering 123 by naming the worksheet file containing the macro 'AUTO123.WKS' and naming the macro '\0'. Procedure 'OUTOLOT' simply runs a batch file from DBASE which calls '123'. An option was also available for the user to run NPV calculations for other crops and LU's within the LOTUS environment before returning to dBASE III. It would be a simple matter to allow the user to alter the discount factor from the menu and so run a sensitivity analysis, for the land unit and crop of interest. The answer from the financial analysis is then exported to a file called DBSMES.PRN from '123'. The 'PVDISPLAY' procedure opens a database called 'PVANSWER.DBF and appends the answer for the financial calculation from the 'PRN' file exported from Lotus 1-2-3. Apart from the slight delay in going through the DOS operating system using a batch file, the use of another software in combination with DBASE was not self evident to the 'naive' user. The financial answer was displayed by use of the '@ - -SAY' commands in a similar manner to that already illustrated (see screen dumps in appendix D).

5.5.2.6 The Macro

In Lotus 1-2-3 a macro is a set of instructions made up of a sequence of keystrokes and commands which are typed into a work sheet as cell entries. The macro is assigned a name and when that name is typed 1-2-3 reads the instructions and performs the specified tasks. If the macro is given the special name '\0' then that macro will automatically be executed when the worksheet is first opened. The complete listing of the macro is given in appendix D. Fig 5.15 is a flow diagram describing the macro, figure 5.16 illustrates the tabular representation.

Fig 5.15 Flow diagram explaining the structure of the Lotus macro for calculation of Net Present Value.



Macro called and onscreen explanation given.

CHOOSE routine
Imports file containing message from DBASE III detailing crop and LU. Selects the data file name, corresponding to the crop and LU of interest.

SELECT routine
Reads in the relevant data file.

PVBEN routine
Calculates the present value of benefits. Calculations were made for 1 year and then replicated for all years by copying formulae to remaining cells. The average PV over the period for the crop is then calculated.

Return control to CHOOSE routines
Display answer and clear crop/LU specific data, write out answer to .PRN file for appending into the the DBASE system, quit worksheet and 1-2-3, returning control to batch file and DBASE III.

Figure 5.16 Showing the tabulation of data within the Lotus macro, extracted from data on coffee in Rayong.

Table 1 Showing amounts/rai for input/output items.

Year	Seedlings	Chemical fertilizer	Family labour	Tractor input	Production
	No.	Kg	md.	hrs.	Kg
1	240	300	125	7	0.00
2	0	350	125	0	0.00
3	0	400	125	0	350.00
4	0	420	125	0	430.00
"	"	"	"	"	"
"	"	"	"	"	"
30	0	400	125	0	300.00

Table 2 Showing price/unit for input/output items in production with a discount factor=12% (Baht/unit).

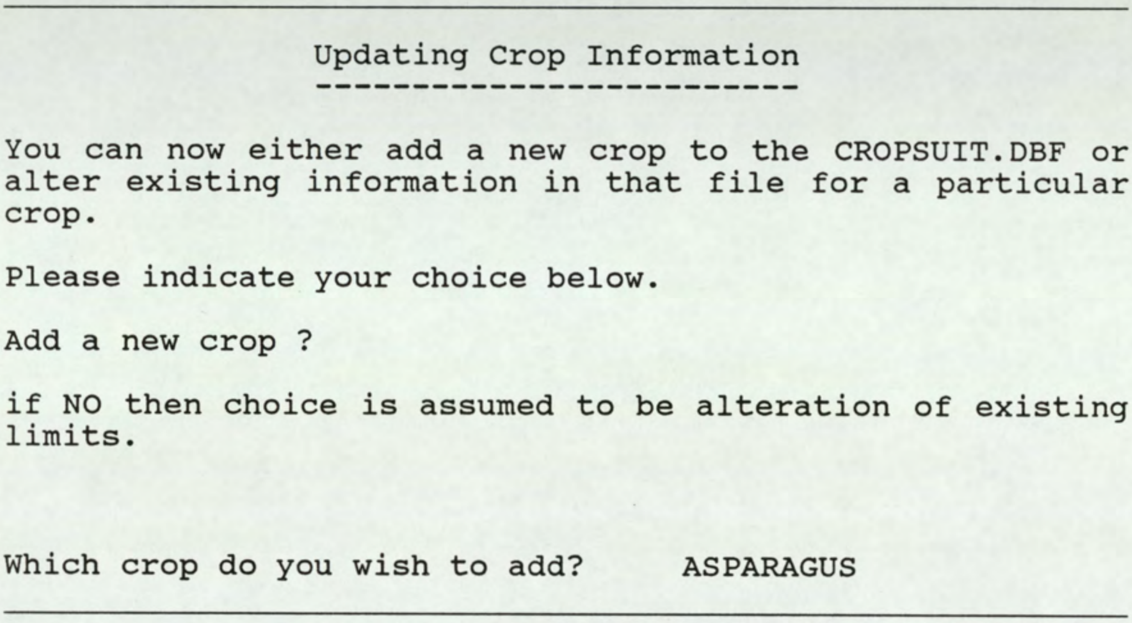
Year	Seedlings	Chemical fertilizer	Family labour	Tractor hire	Production value
1	0.89	5.36	35.72	18.75	19.65
2	0.00	4.78	31.88	0.00	17.53
3	0.00	4.27	28.48	0.00	16.66
4	0.00	3.82	25.44	0.00	13.99
"	"	"	"	"	"
"	"	"	"	"	"
30	0.00	0.20	1.32	0.00	0.73

The columns in table 1 were multiplied by the complementary costs and prices in table 2 and the resulting input and output costs/benefits over thirty years were summed. The Net Present Value is then simply the Total Output Value - Total Input Value. An average figure obtained by dividing by the total period was presented, for comparison with annual crops.

5.5.2.7 Updating the Knowledge Base

The procedure 'ALTREQ' was written to allow a user to alter the information on a new crop. Memory variables were used to store user choices, made from menus designed with the '@- -SAY' and 'GET' commands, as shown in the screen dump in figure 5.17. If a new crop is to be added, the screen message shown in figure 5.18 is displayed and the information is appended at the end of the CROPSUIT.DBF, which is opened unindexed. If alternatively, information on an existing crop is to be updated, then the database is opened with an index on the 'CROP' and 'Rating' fields so that the correct records can be located. In either case the 'GETINFO' procedure is then used to add or alter values of the relevant attributes.

Figure 5.17 Screen dump showing the display for alteration of crop information.



```
                      Updating Crop Information
                      -------------------------

You can now either add a new crop to the CROPSUIT.DBF or
alter existing information in that file for a particular
crop.

Please indicate your choice below.

Add a new crop ?

if NO then choice is assumed to be alteration of existing
limits.


Which crop do you wish to add?     ASPARAGUS
```

The 'GETINFO' procedure makes extensive use of the 'PICTURE' option, to restrict the data that may be entered for a variable. This was important because if data was entered in the incorrect format the matcher procedure could fail, because of a difference in case perhaps, or the use of a letter instead of a number. Such safeguards also protect against unintended mistakes such as miskeying.

Figure 5.18 Showing the format of the screen on choosing to add information on a new crop.

```
                              NEW CROP
                              --------
For each quality, enter all possible class values for the
crop at each level of suitability i.e. S1,S2,S3 or S4
(Equivalent to NS).

    No commas or spaces are requires between each value.

EXAMPLE:  For RUBBER at the S2 level

          Soil surface Texture may be Coarse loamy(2) or
          Fine silty(5) of Sandy(1).
          This would be recorded as 125 or 251 or 521 etc.

It is very important that all possible classes are
covered, as if not a matching error may occur in the
evaluation.  In other words for a land unit where the
missing class occurs, evaluation will give an error.

Press any key to continue...


ASPARAGUS                          RATING    S1
DRAINAGE CLASS                     SLOPE CLASS
SOIL SURFACE TEXTURE               SOIL SUBSURFACE TEXTURE
SOIL DEPTH                         AGROCLIMATIC DIVISION
SURFACE CEC                        SUBSURFACE CEC
SURFACE pH                         SUBSURFACE pH
(water)
SURFACE POTASSIUM                  SURFACE PHOSPHATE
```

5.6 Prolog

5.6.1 Program Structure

Prolog lends itself to a top down design to programming, which is the approach now widely recognised as improving the efficiency of programming. Prolog is especially suited to this technique because of it's goal orientated structure (see section 4.7.3). To resolve the first goal in a program the sub-goals to the right of an 'If' in a logical statement must be resolved. Each sub-goal will in turn depend on other sub-goals. A hierarchy of deductive clauses can be developed by making the sub-goals of a clause, the goals of other clauses lower in the hierarchy. This is illustrated by the pseudo-code and diagram in figure 5.19, which approximates to the logic for some of the rules concerning durian in the Prolog program written for this research. The sub-goals are resolved in a depth-first manner and left to right order, as indicated by the numeric coding in the decision tree diagram.

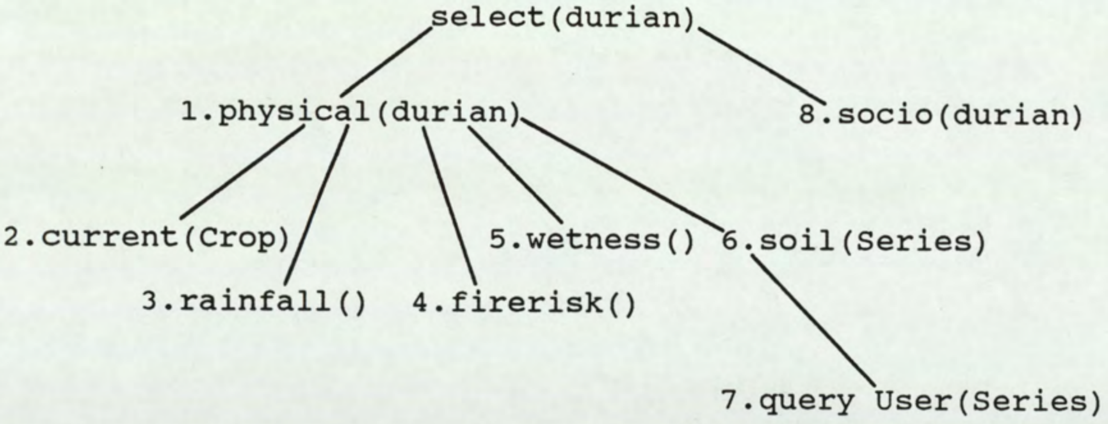
In that program structure is declarative rather than procedural, Prolog differs in a crucial respect from the expert system shells discussed previously and the database system developed. As was explained in section 3.4.5, in Prolog the data is the program, with very few procedural constraints imposed explicitly. Thus the Prolog system designed consisted at the 'top level' of sets of rules determining whether different crops would be suited to the land under consideration. All crops relevant to Rayong province were considered. The crop rules call other lower level rules to query the user on pertinent factors. In this sense there is a hierarchical relationship between crop rules and lower level rules for particular factors, but no hierarchical relationship is assumed for the crop rules, the possibility of growing each and every crop is tested for each query session. This is important because it has a radical effect on the appropriate programming

Figure 5.19 An illustration of how top-down program design occurs naturally in design of a Prolog program.

Logical clauses for decision on growing durian:

```
select(durian) IF  physical conditions for durian are met
               AND socio-economic conditions are correct.

physical conditions for durian are met IF
               current crop is not a high value tree crop
               AND
               rainfall is not of high intensity
               AND
               there is no fire risk
               AND
               the land does not flood
               AND
               the soil is suitable.

the soil is suitable IF
               the soil series is confirmed as appropriate
```

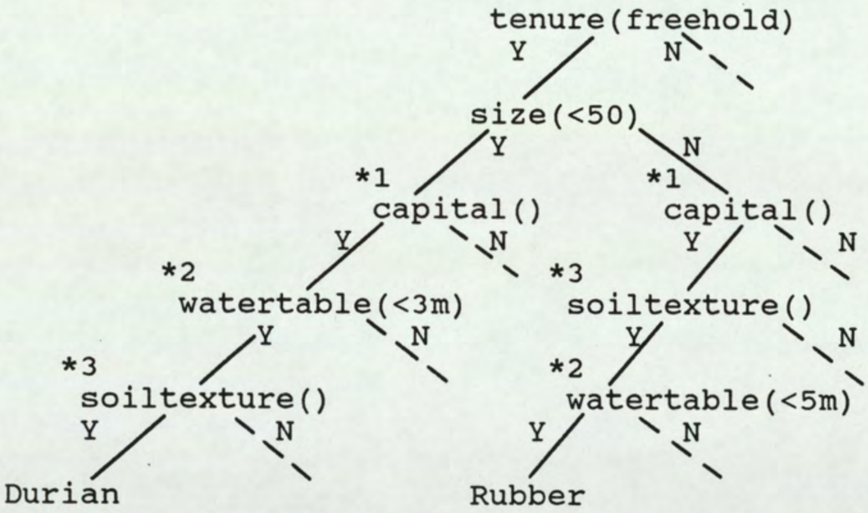
Given below is the hierarchical tree representing the relationship between the above clauses (each node representing a goal predicate to be resolved):



The numeric code for each sub-goal indicates the order of execution, beginning with 1.

structure. Because the decision to grow different crops is based on a similar set of attributes in each case and all crop options are explored, applying a classical Prolog approach would result in question duplication see fig 5.20. In other words information on a particular factor would be sought as many times as that factor appeared in different crop rules. This is due to the fact that all intermediate information is lost on backtracking in Prolog, so on exploring each new branch of a hierarchical tree, information gathered previously is lost. Once a goal branch in a decision tree is explored, information used in resolving the situation is ignored. Purists may argue that this is consistent with finding a logical resolution to a goal, as the goals identified are the only objective. In practice this is no problem where there is a unique solution, or when some kind of hierarchical structure obtains, but where several valid solutions exist a classical implementation is impractical. A flow diagram for the Prolog program written (called FARMTHAI.PRO) is given in figure 5.21, the full listing is given in appendix E.

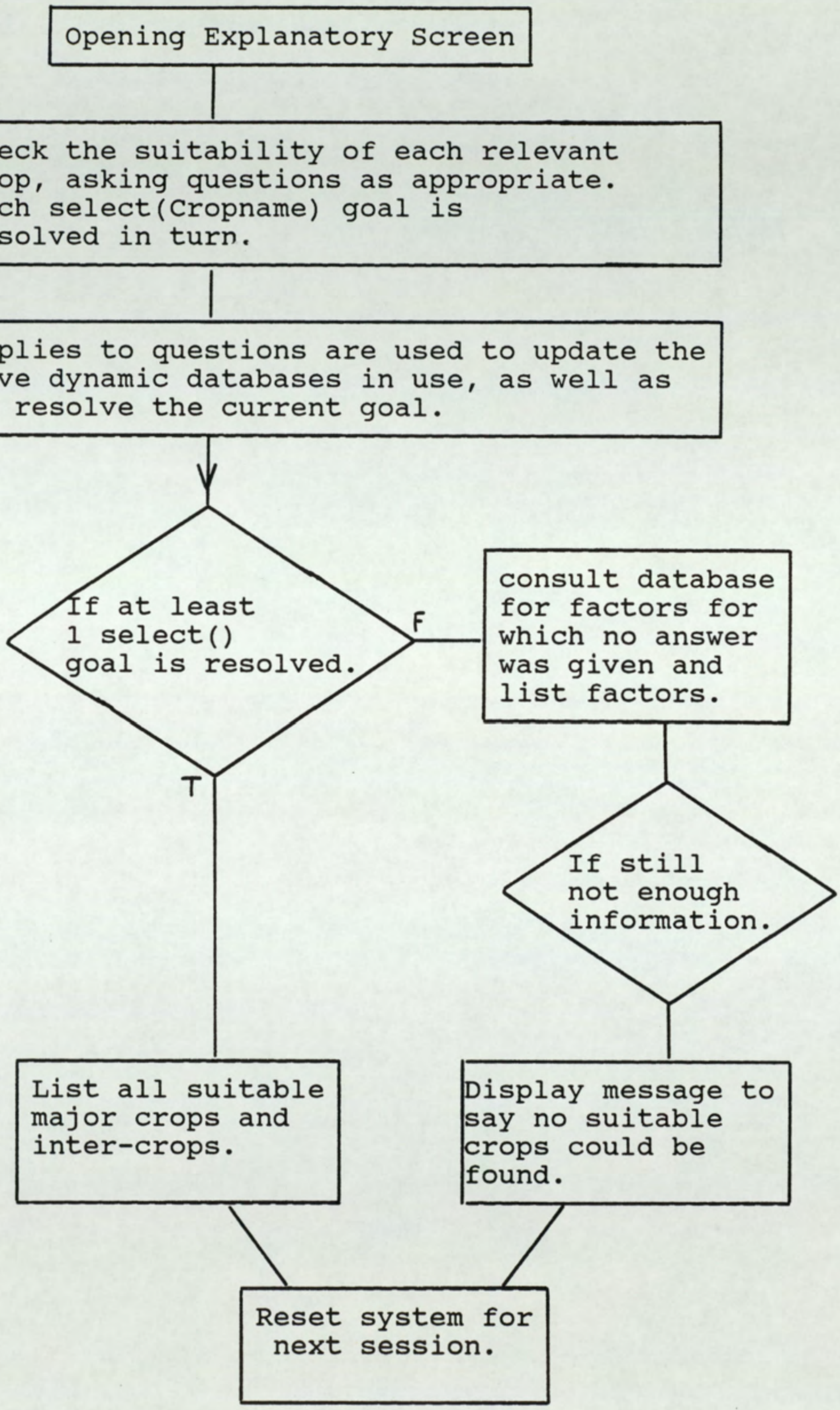
Figure 5.20 Simplified diagram showing the problems of duplication of questions in a hierarchical design.



Duplications: *

The crop choice problem can be thought of as tabular in structure with rows of similar attribute sets checked for each possible crop, against the attributes of the land (as was shown in figure 1.1 in chapter I). When evaluating a particular land unit, what is required is information on each attribute to be stored as it is given, so that it can

Figure 5.21 A plan of the Prolog program, FARMTHAI.PRO, to run a crop selection system for Rayong province, Thailand.


Opening Explanatory Screen
Check the suitability of each relevant crop, asking questions as appropriate. Each select(Cropname) goal is resolved in turn.
Replies to questions are used to update the five dynamic databases in use, as well as to resolve the current goal.
If at least 1 select() goal is resolved.
F
T
consult database for factors for which no answer was given and list factors.
If still not enough information.
List all suitable major crops and inter-crops.
Display message to say no suitable crops could be found.
Reset system for next session.

then be referred to in checking each crop, without repeating questions. This necessity stems mainly from the independence of attributes identified as important and the fact that attribute sets for each crop overlap.

The method of programming had to deviate somewhat from the 'Edinburgh Standard ' prolog because of the need to store information. Fortunately most Prolog implementations allow for use of a database of information, by provision of the standard predicates assert() and retract(). A dynamic database in turbo prolog is a database to which facts can be added during execution. Manipulation of the database depends upon the use of 'assert' and 'retract', to alter the database, by adding or removing facts respectively. Prolog purists tend to disdain the use of 'assert' and 'retract' and are chary of the concept of the user altering the program. Yet it is precisely this kind of flexibility, that is demanded if an automated land evaluation system is to be easily updated. Facts which are 'asserted' to the database domain (which may include more than one database predicate) are maintained and can be verified, unless they are retracted.

If there was only one goal to be verified (choice of a single crop for example), then it might be possible to avoid using assert and retract, because repetition of questions need not occur, as all questions could be asked as part of a single goal clause. The problem is seldom so closely defined though and such procedural care is in contradiction to the ethos and benefits of programming in Prolog.

The use of a database does not detract from the main feature of programming in Prolog, which is its declarative style, database predicates are treated the same as any other predicate, although restrictions on domain types are slightly more severe, as multiple database predicate declarations are not allowed. This simply necessitates use of more predicates and thus results in a slightly longer program but with no effect on performance. In pure Prolog no predicate declarations are required which simplifies the application of predicates, but in Turbo Prolog acceptable arguments for a predicate must be declared.

At the declaration stage, some thought must be given as to how much categorization of data can be expected of the user, e.g. can original slope values be accepted as integers, or should they be categorised first. Either approach can be handled easily and decisions of this kind on data structure, are handled at the pre-design stage, when the objectives and aims of the proposed system should be specified. Most prolog implementations are not suited to handling real (decimal) numbers, but Turbo Prolog does have standard predicates to achieve this, for the simple comparisons normally required in land classification this does not present a problem.

Logical terms may be rules, facts or queries. A query is actually just a specialised form of fact, to be resolved as true or false (see section 3.4.4). To meet the objective of establishing which crops are suitable for the land, necessitated specification of the important requirements for the LUT's concerned, the aim was to compare the approach with other methods, in particular expert system shells. For this reason the same baseline data was used as in the Super Expert system (see section 4.4.2.3). The same questionnaires were used, but instead of using values for the pre-defined attributes as with the expert system shell, the detailed comments made by farmers in answer to questions on why they were, or were not growing certain crops, were used to formulate rules based on farmer experience. Some discretion in formulating rules had to be exercised by the author based on farmer experience, chiefly to make explicit some facts taken for granted by the farmer, but there was minimal interference with the rule set obtained from the farmer. This means omissions and other errors in the simple system developed are likely, but the intention was to compare methods in terms of knowledge representation and efficiency. Without returning to the same area, to work with more farmers, development of a completely reliable system would not be possible. In building knowledge based systems by whatever means, iterative testing and development is essential.

Having established rules for each crop these were simply written into the program virtually as they would be expressed in natural language, for instance the following is syntactically correct:

```
crop-is ("Rubber") If
     land-is (freehold) and credit-is (good) and size>50.
```

It is at this stage in the development of a Prolog system that the natural language style is particularly useful. Taken to the extreme, rules can be written into a Program as they are expressed by the farmer, see section 5.6.3. As the low level predicates for input and output to the user are decided, the rule predicates may then be changed to fit a certain format, but the meanings of the predicate and it's arguments are easily maintained, so that the logical structure is still obvious to the programmer and user alike.

5.6.1.1 Input/output

Once crop rules were entered, the next stage in the top-down process, was to decide how the information on each attribute would be gleaned from the user. Data types were already identified at the stage of defining objectives, so these were declared and this influenced which standard predicates were available for input and output. The next decision was, how much option to give the user. This is the other stage in prolog programming where many people stress the advantage of a natural language interface, whereby the user can simply enter his answer to a query, in the way he would usually express himself. In practice this is difficult to implement in a system of any size. As long as the queries can be satisfied by yes or no answers there is little that can go wrong, but give the user more latitude to use his imagination and an unknown value or expression, may be entered. In Prolog there will not normally be any indication that the value was out of range or inappropriate, the goal will simply fail and in this case a feasible crop would not appear as an option.

In essence, in this situation the user is asking the program to accept a fact which is not established. Prolog is designed to arrive at a logical proof given certain facts. All attribute values for which solutions are given in the rule base, are assumed to be established facts, if confirmed by the user. A value which does not occur within the rule base cannot be established as a given fact, so the resolution of the query is bound to fail. If the reply was actually valid, then during the iterative 'training' of the system, such values should be included in the rule base, but to take account of every variation on an answer, where say sentences were allowed, would seem to be very difficult. It is also usually unnecessary in the classification problem, since sensible options have normally been identified and can be offered as choices.

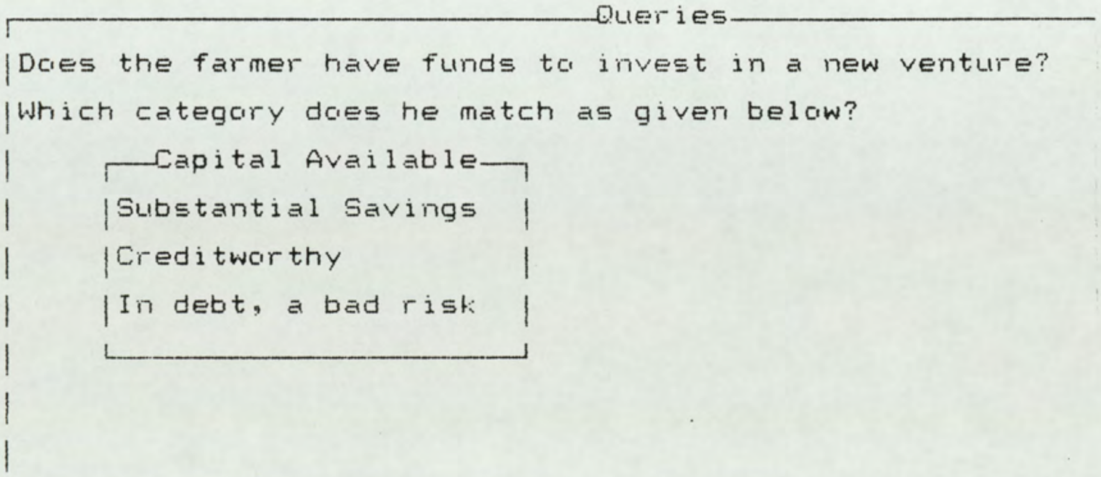
Mis-spelling of replies, or language alternatives, can be provided for reasonably easily, since there is a fairly limited domain of possible answers. This was done as an example in the Prolog system developed (see Appendix E), for a few of the queries used. Replies were checked for their legitimacy, first against a list of all different valid answers and then against lists of synonyms for each valid answer, if the first check failed. The code excerpt for this is shown in figure 5.22.

Figure 5.22 Code from FARMTHAI.PRO showing one means of checking the legitimacy of answers and ensuring the correct name is matched.

```
legitimate(Ans,Crop):-
   member(Ans,["Paddy rice","Cassava","Durian",     /*This predicate checks that*/
   "Para rubber","Coconut","Rambutan","Mangosteen", /*the answer is one of the*/
   "Coffee","Sugar cane","Sweet tamarind","Pineapple",/*crops listed between the */
   "Banana","groundnuts","Mungbean","Chilli",       /*square brackets*/
   "Jackfruit","Lamud","Mango","Eucalyptus",
   "Casuarina","Cashew","Vegetables","none"]),!,
   Crop=Ans.

legitimate(Ans,Crop):-   /*This predicate is called if that above*/
   synonym(Ans,Crop).    /*fails.  The answer is then checked by */
                         /*synonym predicates against other names*/
synonym(Name,"Para rubber"):-      /*for crops, known to be used */
   member(Name,["Rubber","rubber","pararubber",   /*and common mis-spellings*/
   "para rubber","Ruber","ruber","rubbe"]),!.

synonym(Name,"Durian"):-
   member(Name,["durian","DURIAN","TOOREAN"]),!.
```

The development of memory resident spelling checkers and thesauruses allows for the possibility of real time verification of a sentence, by accessing such dictionary and thesaurus files. This would be one possibility for improving the natural language interface, by what is essentially a simple list processing procedure, previously very demanding on time but now very fast even on a desk top micro.

Figure 5.23 Showing a screen dump of the query on capital.



```
┌──────────────────────────────Queries──────────────────────────────
│Does the farmer have funds to invest in a new venture?
│Which category does he match as given below?
│      ┌──Capital Available──┐
│      │Substantial Savings  │
│      │Creditworthy         │
│      │In debt, a bad risk  │
│      └─────────────────────┘
│
│
```

Where possible options are not in doubt, but there is uncertainty as to whether the user will be aware of all the choices, a 'pop up menu' is probably most appropriate, with selection by use of the cursor keys. This can be easily arranged with appropriate programming tools, as shown in the screen dump in figure 5.23. The query on capital available for investment in a new enterprise, is an example of this kind of method used in the system. For integer values, range checks can easily be arranged and this can also be done for ordinal alphabetic characters, by conversion to the corresponding ASCII integer value.

The precise low level structure of predicates for input and output, will vary according to programmer preference, or user demand but the program written and described here, shows that a 'Friendly' interface can easily be created in a short time.

5.6.2 Procedural control

Once the rules have been entered and the input/output format designed, the program is essentially complete. Note there was no requirement for if - -then - -else; or while--DO--; or CASE constructs, as commonly used to impose the required sequence of operations in other languages. This is because of the Prolog declarative structure, whereby control is avoided. Only the required clauses will be used and this is determined by the initial 'Goal' statement, in conjunction with the query answers supplied.

In fact, for the sake of efficiency and to maintain an easily understood and sensible output, some minimal control is imposed. This comes from the use of the 'cut' predicate to limit resolution of a goal to the first successful clause found, thus preventing backtracking providing multiple solutions to a goal where this is not required, as this may be wasteful of time and could lead to an answer being output several times, equivalent to the number of times the goal has succeeded. The 'cut' written as '!' is provided as a built-in system predicate, to be used to prevent Prolog following such fruitless computation paths.

To avoid any compromise on the logic of the program, the use of cut was minimised in the land use planning program, but it proved particularly useful in disjunctive situations where a goal could be resolved by a number of alternative conditions, as for example in the case of selecting cassava as a suitable crop, shown in fig 5.24. The socio ("Cassava") clauses are not mutually exclusive. In this case it would look odd to the user to be given the answer cassava twice, when he is only asking about one piece of land. The cut at the end of the 'socio("Cassava")' clauses, ensures that once this point is reached no further solutions will be sought and cut always succeeds. So that if the goal succeeds once, it will not be tried again.

Figure 5.24 Showing a code excerpt from the Prolog program illustrating the use of the cut.

```
                                                        /*EXPLANATION*/
socio("Cassava"):-
   affirmative("Is cashflow important? y/n or ?",cashflow),  /*If cashflow is */
   determine(capital,X),                                /*important, the farmer*/
   X=3,                                                 /*has debts, sophisticated*/
   affirmative("Is ease of management important? y/n",easy_manage),  /*management*/
   affirmative("Is family labour available? y/n",famlab),!.   /*cannot be ensured*/
                                   /*but family labour is available, grow Cassava*/

                                   /*This represents an alternative*/
socio("Cassava"):-                 /*situation under which cassava*/
   confirm(current,"Cassava"),     /*would be grown, because it has*/
   determine(capital,X),           /*been the previous choice, the*/
   X=3,                            /*farm is small and no risk can*/
   determine(size,Y),              /*be taken.                    */
   Y<5,
   negative("Can the risk of crop failure be tolerated? y/n",risk),!.

socio("Cassava"):-
   negative("Does the farmer possess the freehold or long term tenure
   on his land? y/n",tenure).      /*Where the farmer has no permanent tenure*/
                                   /*Cassava is one of the few options.*/
```

Another means of control used in the program was fail. This offers a method akin to a sub-routine, to output a set of solutions at a particular point in the evaluation, where if this was not used the whole clause would have to be repeated again, which may not be desirable and indeed may be prevented by a later cut. The use of cut and fail in combination can be very effective. The use of fail is shown by the code fragments given in figure 5.25. The unknown factors should all be listed by the write() predicate, but without fail this would be difficult to achieve if 'noinfo' was called once only. The first answer would be returned but the other factors would not be checked (presuming the calling predicate contained a cut).The list of factors for which no answer was given was held against the predicate 'unknown()'. This can be envisaged explicitly as a list of clauses for the predicate 'unknown()', such as:

```
unknown (current-crop)
unknown (water-depth)
unknown (tenure)
```

Figure 5.25 Showing a code excerpt from the Prolog program illustrating the use of the 'fail' predicate.

```
noinfo:-             /*Writes out a list of factors for */
        unknown(Y),  /*which no information was available*/
        write(Y),nl,
        fail.

noinfo.
```

The predicate 'noinfo()' was used to unify with the first unknown clause and write out the argument name e.g. 'current-crop'. The fail predicate then stops the 'noinfo()' clause succeeding and causes backtracking to establish if there are alternative unifications for the 'unknown(Y)' predicate, this causes 'water-depth' to be written out and so on. When no more unifications for 'unknown(Y)' are possible the first 'noinfo()' goal fails, but to avoid failure of the run predicate a second 'noinfo()' clause, with no conditions follows to ensure successful resolution, the goal then continues.

The final procedural control to be found in the crop choice program is inherent to Prolog but perhaps not immediately obvious. Clause ordering is very important. Clauses for the same predicate are tested for unification in sequential order down the program listing and clauses for the same predicate must be grouped together. Thus in the crop choice program when the goal is 'select(crop)', the select clauses are tested in order of appearance, which in this case is: paddy rice; cassava; durian; para rubber, and so on. For logic programming purists even this seemingly negligible control is frustrating, but very difficult to avoid. It is frustrating because for the pure logic model the ordering of clauses should be irrelevant, for the same predicate, no one clause should be selected in preference to another. However, the non-deterministic choice of a clause is very difficult to achieve in practice.

Figure 5.26 Showing a code excerpt from the Prolog program illustrating how questions can be asked unnecessarily

```
physicalessentials("Durian"):-
  confirm(current,Crop),
  Tree=Crop,
  not(member(Tree,["Cashew","Coffee","Para rubber","Tamarind"])),
  negative("Is the wet season rainfall of high intensity? y/n or ?",
  raindrops),
  negative("Does nearby bush present a fire risk? y/n or ?",
  fire_risk),
  not(condition(paddy_land)).
```

The consequence of this control in the case of crop selection, is that some questions may be asked unnecessarily. The code fragment shown in figure 5.26 illustrates this for the fruit tree crop durian. The suitability of the land would have already been checked for paddy rice and cassava, the code shown is the first set of clauses to be checked to establish suitability for durian. The land may have proved suitable for paddy rice, but until the last question on paddy land this would not be checked. The two crops are incompatible in there conditions, but questions on three different attributes concerning durian (namely: current crop, rainfall intensity and fire risk) would have been asked unnecessarily. The obvious solution is to move the last clause up, to be the first sub-goal in satisfying the 'physicalessentials()' predicate. With care taken in the ordering of clauses in the body of a goal, inclusion of checks on previous database predicates and ordering of goal clauses themselves, programming can be arranged to avoid needless questioning which mystifies the user. It should be recognised that this then requires serious thought on the procedure to adopt and this becomes very complicated in large programs. The great advantage of declarative style is steadily eroded as more thought has to be devoted to procedure. It would seem there is a pay-off between the advantages of fast and logically valid programming, as against erring to the more contrived, but apparently consistent programming style, based on a closely followed procedure. Undoubtedly it is a high priority to avoid annoying the user with unnecessary and apparently stupid questions from a supposedly intelligent system, but compare this with a database system where it is much more difficult to avoid asking the same complete question set on every query session. In Prolog most questions are pertinent and every session differs according to the answers supplied. Much of the unnecessary questioning can be remedied on training of the system.

If the inherent procedure of clause order is not dwelt on, then once the odd cut or fail has been added to the rules, the program is complete. Very rapid development from the original information can be achieved by building straight from the information itself. The Prolog system written as FARMTHAI.PRO, completed to the stage shown in the listing given in appendix E, took 62 man hours to write.

5.6.3 Using Prolog as an Interview Tool

The preceding sections have described how Prolog was used to try and rapidly achieve a prototype model of typical farmer's decision making, in choosing which crop to grow. The hypothesis was that by use of a declarative as opposed to a procedural language, development time would be reduced and the implementation achieved would be an improvement, in terms of ease of use and comprehension. Another means of applying the Prolog language to reduce development time was also investigated. The approach was to simulate the use of Prolog as an interview tool. This meant keying in farmer replies to questions directly from questionnaires, as though they were being given verbally. Ideally the technique should have been tested in the field with farmers personally, but at that time no portable computer was available.

Information from farmer replies was keyed straight into a Prolog program in Prolog clause form, over a period of one hour. This period was typical of the interview time taken with a cooperative farmer in Thailand. In this time eight questionnaires were processed. Only questions 5,6 and 7 of the questionnaires were consulted (see appendix I), concerning why different crops were grown or not. If the approach was used in a real situation most information would normally be gathered by informal discussion with the farmer. The thirteen clauses entered and notes made over the period of one hour, are typical of the diversity and amount of information likely to be supplied by a farmer. Some of the information entered is shown in figure 5.2.7, the full text is given in appendix E.

It can be seen that predicate names and arguments can easily be chosen to convey the original meaning of a farmer's remark e.g. 'marketDistance(pineapple,Dist)'.

Figure 5.27 An example of the way information can be recorded directly from the farmer.

```
socio("pineapple"):-
            determine(capital,A),
            marketdistance(pineapple,Dist),
            Dist<20,
            confirm(experience,pineapple),
            confirm(labour,availablecheap),  /*expensive hired labour is*/
            affirmative(_,highmanagement),   /*a constraint.*/
            confirm(availability,pineappleSuckers),

socio("groundnut"):-
            confirm(current,C),
            member(C,"Para Rubber","Duri[illegible]","none"),

socio(cassava):-
            affirmative(_,familylabour),
            negative(_,highmanagement),      /*Not interested in crop requiring*/
                                             /*high management input*/
```

A problem that arises as a system is built is the use of new predicates at an interview, when an existing predicate would cover the situation. This kind of mistake would seem inevitable, but as long as the meaning is self evident, alterations can easily be made on incorporation of the new predicates in the existing system. Where the meaning might be misconstrued, notes can be added alongside or at the end of the listing (between Prolog quote marks /*- -*/). The other problem that arises is that most rules are not regarded as complete, as each farmer only supplies limited information specific to his own circumstances. So each rule represents only a partial example. The danger with partial examples representing situations where a crop should be grown, is that if applied without modification, constraints might be ignored. One approach is to group the partial rules until a complete rule is formed, such as that shown below for socio-economic factors concerning cassava.

```
      socio("Cassava"):-
            affirmative("Is cashflow important?",cashflow),
            determine(capital,X),
            X=3,
            affirmative("Is ease of management important?,
            easy_manage),
            affirmative("Is family labour available?,
            famlab),!.
```

Attention must then be given to the correct grouping of factors, as certain factors only apply under particular circumstances and under a different situation other factors or constraints become important. For example the rule above represents the situation where the farmer has no savings or borrowing capacity; no alternative income so cashflow is important; does not have the ability, or desire to grow a crop requiring intensive management; and has family labour available. The second rule for cassava, shown below, represents an alternative situation where the farm is small in size; capital is still not available and a risk cannot be taken; coupled with the inertia of an existing system of growing cassava.

```
     socio("Cassava"):-
          confirm(current,"Cassava"),
          determine(capital,X),
          X=3,
          determine(size,Y),
          Y<5,
          negative("Can the risk of crop failure be
          tolerated?",risk),!.
```

Under the second situation ease of management is unimportant, as growing a crop like vegetables requiring intensive management, would be too risky anyway. Similarly, the issue of family labour availability is of less consequence for a small area of land as labour costs incurred would not be high. The final cassava rule concerns tenure: if there is no permanent tenure then cassava is the most profitable and least risky venture.

```
     socio("Cassava"):-
          negative("Does the farmer possess the freehold
or
          long term tenure on his land?",tenure).
```

The factors in each clause can be thought of as akin to land characteristics as used in a physical classification i.e. they are not independent, but together they define a particular quality. So each socio() clause can be thought of as an independent socio-economic quality. As in a physical classification the correct identification of relevant qualities is an essential task for successful evaluation. Individual factors cannot be considered in isolation.

Having obtained the text in 'near-Prolog' form, from the simulated interview, the clauses had to be incorporated in the main FARMTHAI.PRO program and adjusted to conform. Adjustments to be made included the programming of new rules, to elicit information from the user, of relevance to the new clauses. For instance the rule for querying the user on previous cropping experience of the farmer, had to be added. The adjustments took three hours to complete, with a further hour to debug introduced errors, detected on running the program. On that basis it would take four hours of programming to incorporate the results of one interview into the main program. If all interview information from a day's work was added together, then the debugging time would be proportionally less as it is most closely related to changes in input/output clauses rather than the number of crop rule clauses. Many new crop rules use information from existing query clauses, rather than requiring new queries to be made. A good example is soil requirements. Although these vary with each new crop combination, the information can be collected by one question on the soil series. If six interviews were made in one day, on the above estimate, it would take eighteen hours to build the information into a program. Debugging could in theory take up to a further six hours, but on the basis of the preceding argument that time could reasonably be halved, to give an estimate of twenty one hours programming in total, to automate the knowledge gathered from one day of six interviews.

5.7 Satellite Remote Sensing

5.7.1 Rayong Case Study

For the classification of the MSS imagery training areas were chosen by visual inspection and reference to the DLD map of current land use. Six separate classes could be distinguished which were identified from the DLD map as : Sugar cane; Rubber; Coconut; Pineapple; Cassava; and Paddy rice. All these crops are grown in large expanses, as well as on smaller plots. The training areas on the image coincided with some of the larger areas of each crop shown on the DLD land use map. A second set of areas for each crop was chosen for use in testing the classification obtained. The results of the classification are summarised in the confusion matrix shown in figure 5.28. The raw image data was first geometrically corrected and a contrast stretch applied so that the image could be inspected, as shown in plate 5.1. Plate 5.2 shows the classified image obtained.

	Ground Cover Class	Spectral Class 1	2	3	4	5	6	No.of test pixels	Est.% Class acc.[1]
1	Sugar cane	15	11	37	7	24	7	46	15
2	Rubber	6	48	0	0	24	21	33	48
3	Coconut	16	0	50	16	18	0	68	50
4	Pineapple	3	5	0	71	21	0	66	71
5	Cassava	1	0	9	10	79	0	78	79
6	Paddy rice	1	5	1	0	0	94	182	94

Overall accuracy = 60%

Figure 5.28 A normalised confusion matrix showing the percentage of test pixels accurately classified in the Rayong 1981 Landsat MSS image.

[1] Estimated class accuracy as a percentage.

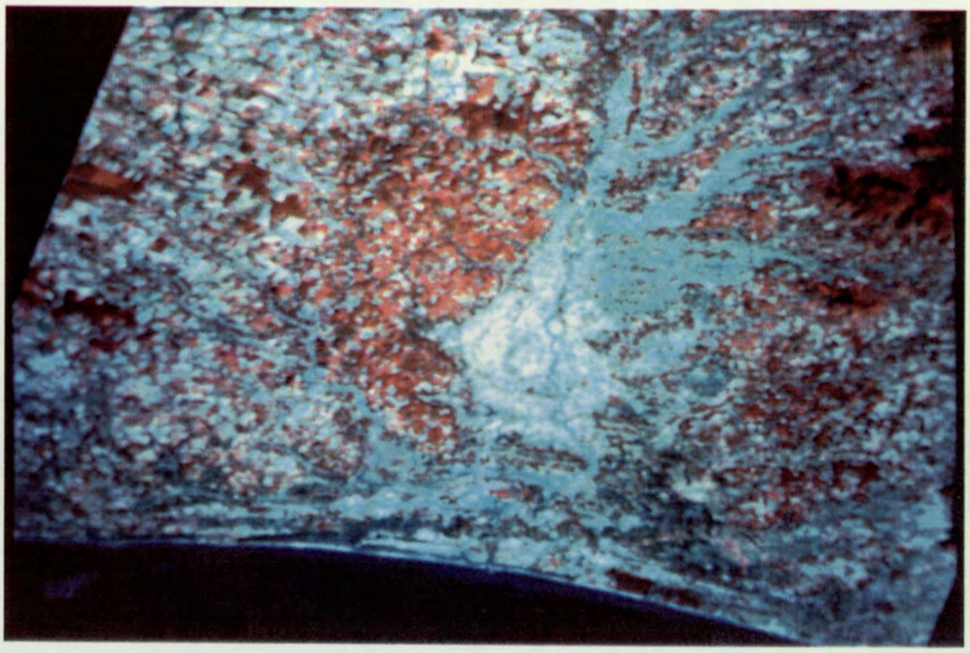


Plate 5.1 Geometrically corrected 512x512 pixel extract of the Landsat image, from December 1981, showing Rayong province with a manual contrast stretch applied.

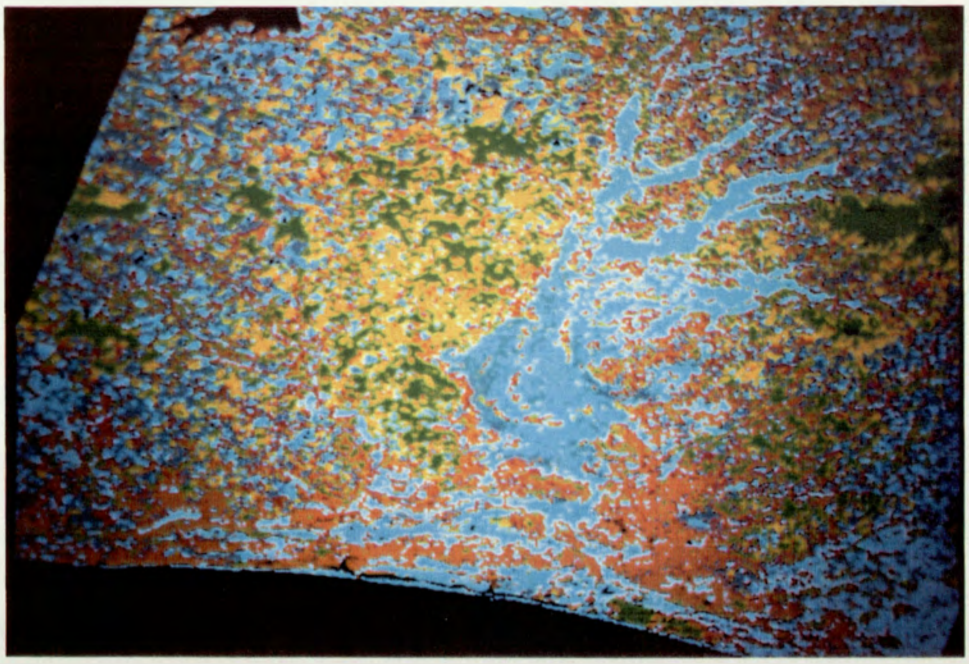


Plate 5.2 Land cover classification of the 1981 Rayong Landsat MSS imagery.

Key

Category	Colour	Category	Colour
Class 1 Sugar cane	Red	Class 4 Pineapple	Cyan
Class 2 Rubber	Green	Class 5 Cassava	Yellow
Class 3 Coconut	Blue	Class 6 Paddy rice	Magenta

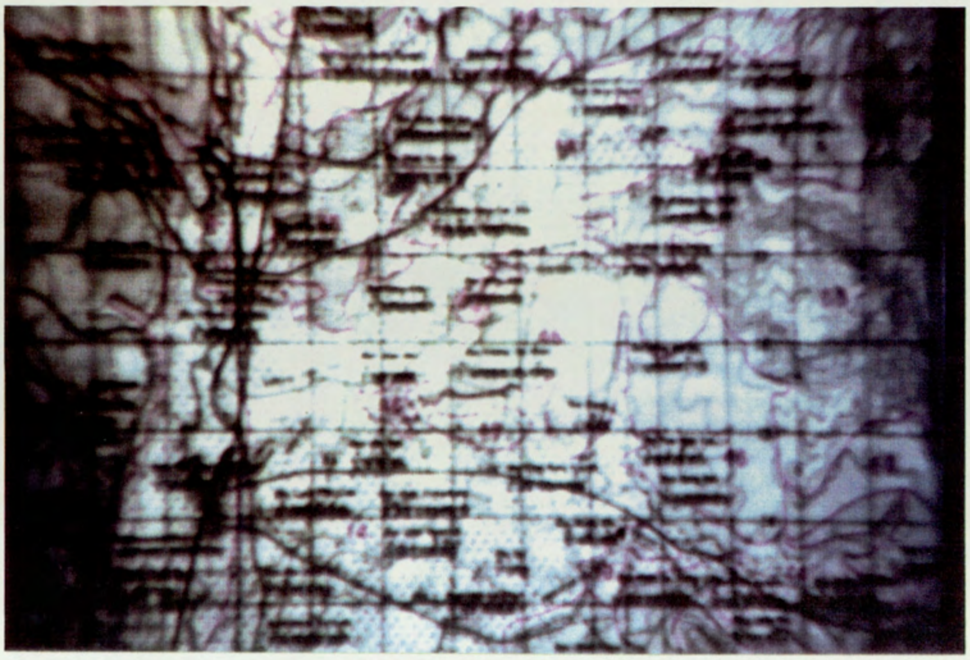


Plate 5.3 A colour composite of the video-grabbed soil map image.

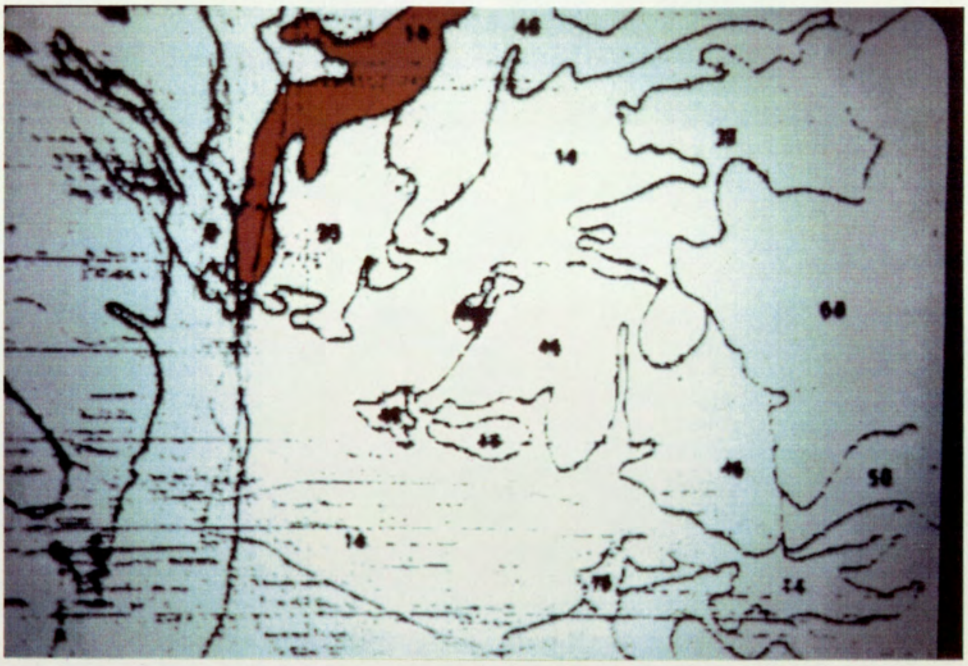


Plate 5.4 The result of an attempt at automatic feature extraction on the video-grabbed image.

As explained in the methodology (section 4.8.2.1), ancillary information on soil type was digitised from the provincial soil map by two different methods, frame-grabbing and by densitometry. The results of the frame-grabbing are shown in plates 5.3 and 5.4.

Plate 5.3 is a composite of the red, green and blue filtered images obtained and plate 5.4 shows the result of attempting to automatically extract the soil boundaries from that image.

Experiments were made with a number of different filters over the video camera lens. Images were grabbed using red; green; blue; cyan; yellow and magenta filters. Different combinations of filtered images were then examined using ratioing and subtraction of different bands, to try and isolate the soil boundary lines from the background detail. It was expected that since the soil boundary lines were printed in magenta, over a background in grey, a magenta filter would give an image with the highest possible reflectance for the soil boundaries. In contrast a cyan filter principally absorbs light in the red band frequency giving very low reflectance values for red lines. It was anticipated that by subtraction of the cyan image from the magenta image, or by ratioing the magenta image over the cyan image, it would be possible to isolate the soil boundaries and then threshold the image. Thresholding an image, by converting all pixel values above a certain value to 255 and all those below that value to zero, results in a binary image showing only the extracted feature. On testing all the likely combinations of filtered images, the ratioing of the red image over the green gave the best result (the same effect was also obtained by subtraction of the green and blue images from the red image, followed by thresholding). The red filtered image actually showed a higher reflectance for boundary lines than did the magenta image. As can be seen from plate 5.4, it was possible to accomplish an automatic fill of one of the soil areas, by giving a seed point from which to start the fill, within the area and filling to the boundary. This was the exception though, as for the other areas incomplete boundary lines resulted in the fill colour 'leaking' to fill the greater part of the screen. Attempts were made using line thickening routines to close gaps in the lines, but none proved wholly successful. The result of the digitisation of the soil map by means of a drum scanning densitometer is shown in plate 5.5.

The image was the result of a broad spectrum band scan, at an aperture of 100 um. The definition in the image is clearly better than that of the video grabbed image. The quality of image obtained from the scanning enabled automatic feature extraction of the complete image to be achieved, by the same ratioing method described above. The result of the extraction is shown in plate 5.6.

The effect of adding soil information to the spectral information from the MSS imagery of Rayong was tested by incorporating information on the extent of paddy soil for amphoe Ban Khai, as explained in section 4.8.2.1 (page 111). Plate 5.7 shows the new classification obtained, with the two additional paddy soil classes identified.

The effect of this additional information on the area statistics is shown in table 5.3. None of the classes in the original classification could represent pure crop categories precisely correlated with ground cover, since all pixels in the image were classified, without imposing a threshold on the minimum distance algorithm. No threshold was imposed because of the nature of the image (see appendix F showing the image statistics and the discussion, section 7.7.1.1).

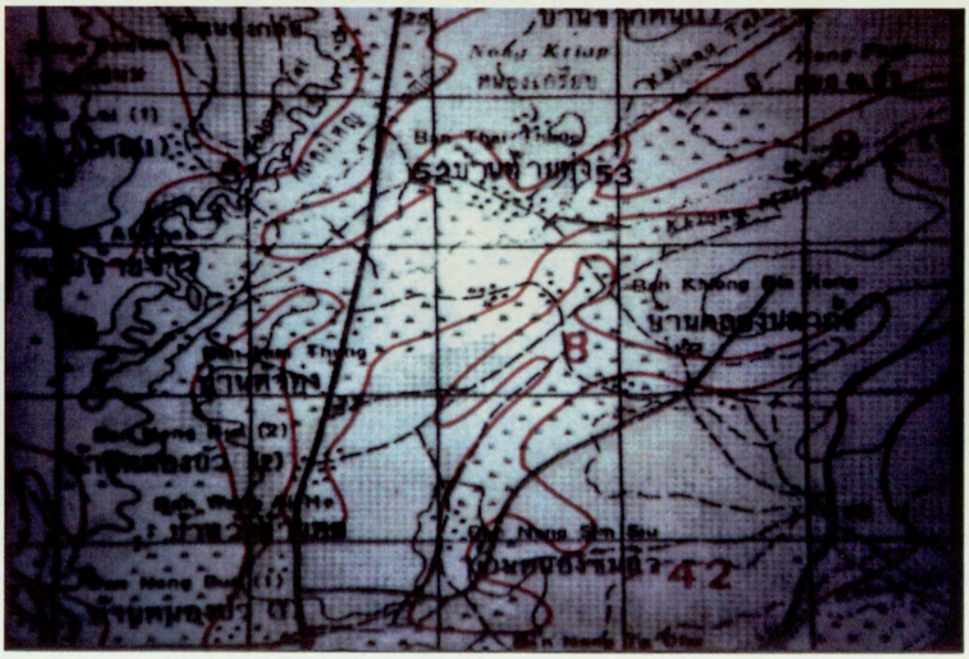


Plate 5.5 Showing the colour composite image of the soil map digitised by a drum scanning densitometer.

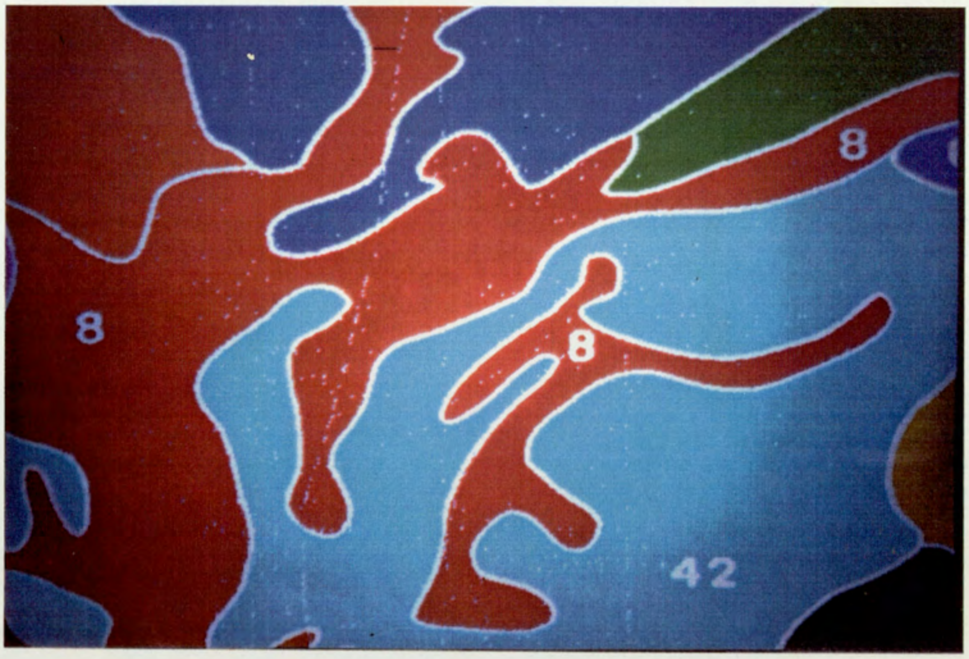


Plate 5.6 Showing the colour coded digital soil map, automatically extracted from the scanned soil map image.

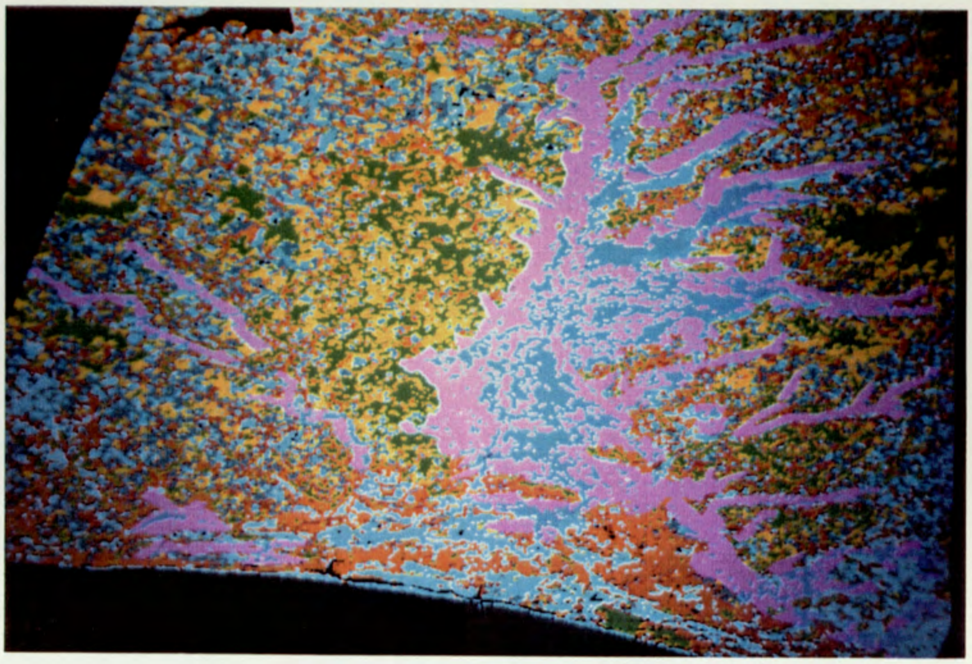


Plate 5.7 The new classification using additional soil information.

Table 5.3 Showing the crop area statistics for Amphoe Ban Khai, with the new classification including soil information and the original spectral classification.

Land Cover Class	Area in Rai(%)		
	A-No soil Information	B-With soil Information	Difference B-A
i) Sugar cane	30,016	20,455	9,561
ii) Rubber	33,508	22,068	11,440
iii) Coconut	20,067	12,777	7,290
iv) Pineapple	21,928	19,339	2,589
v) Cassava	45,450	39,502	5,948
vi) Paddy rice	35,410	13,041	22,369
vii) Paddy rice over paddy soil.		22,369	
viii) Paddy soil not classified as paddy in A.		36,822	

These figures can be compared with the OAE statistics:

			OAE 1978 Survey
Tree crops i) & ii)	53,575(23.5)	34,845(15.3)	31,039(13.6
Field crops i),iv) & v)	97,394(42.8)	79,296(34.8)	60,525(26.6

By overlaying the amphoe boundary on the new classification it was possible to determine the extent of each class within amphoe Ban Khai. The image extract only included 81 % of the amphoe, but the remaining area was of higher ground, including uncultivated hills and did not include any sizeable areas of paddy land. The difference column in table 5.3 shows how the soil information altered the area estimates for each category, reflecting the extent of misclassification. The boundary of the paddy rice area shown on the 1981 DLD current land use map was also digitised and superimposed on the classified image to illustrate the difference in extent of paddy rice between methods. The area within the boundary from the current land use map was also calculated. Plate 5.8 shows the classification with the soil information added, together with the amphoe boundary and DLD paddy rice area superimposed. The area statistics for paddy rice are compared in table 5.4. Table 5.5 shows the 1978 estimates of area, for certain groups of crop types in Amphoe Ban Khai.

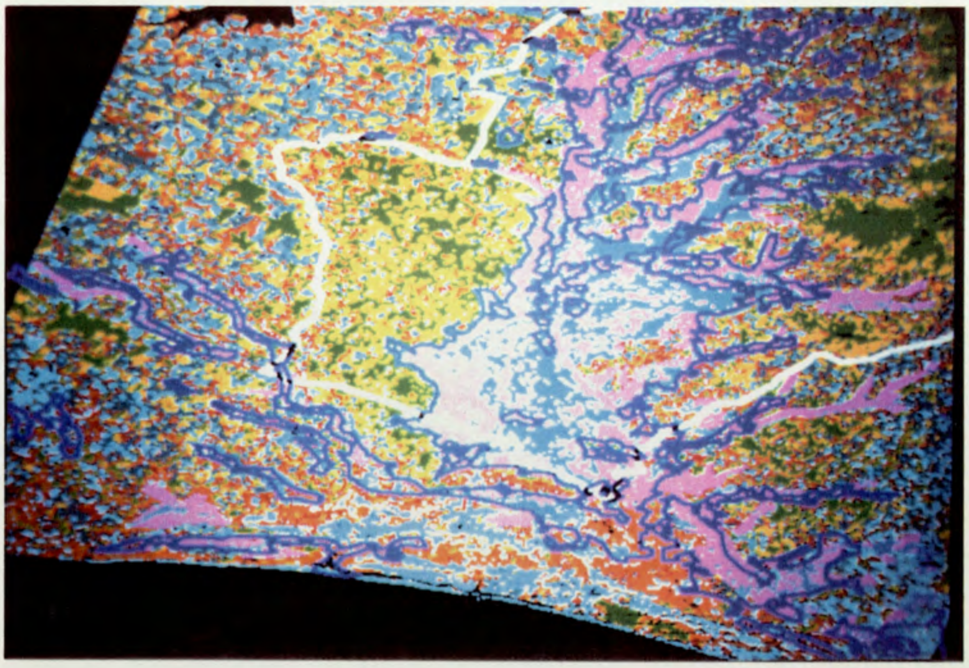


Plate 5.8 Showing the classification with soil information added, together with the amphoe boundary and DLD paddy rice area superimposed.

Area of Paddy rice:	A-Original classification no soil information	B-New classification with soil information	From DLD current land use map	OAE survey 1978
Rai	35,410	72,232	60,594	101,910
% of Amphoe	15.6	31.2	26.6	44.8

Table 5.4 Comparing the different area estimates for cropping of paddy rice, in Amphoe Ban Khai, Rayong.

Crop Type	Rai	%
Rice only	7,309	3.2
Rice followed by another crop.	94,601	41.6
Field/vegetable crops.	60,525	26.6
Tree crops	31,039	13.6
Forest or Pasture	4,507	2.0
Other	29,600	13.0
	227,581	100.0

Table 5.5 The 1978 OAE figures for grouped crop types are given below for comparison with the results from image analysis (in Rai).

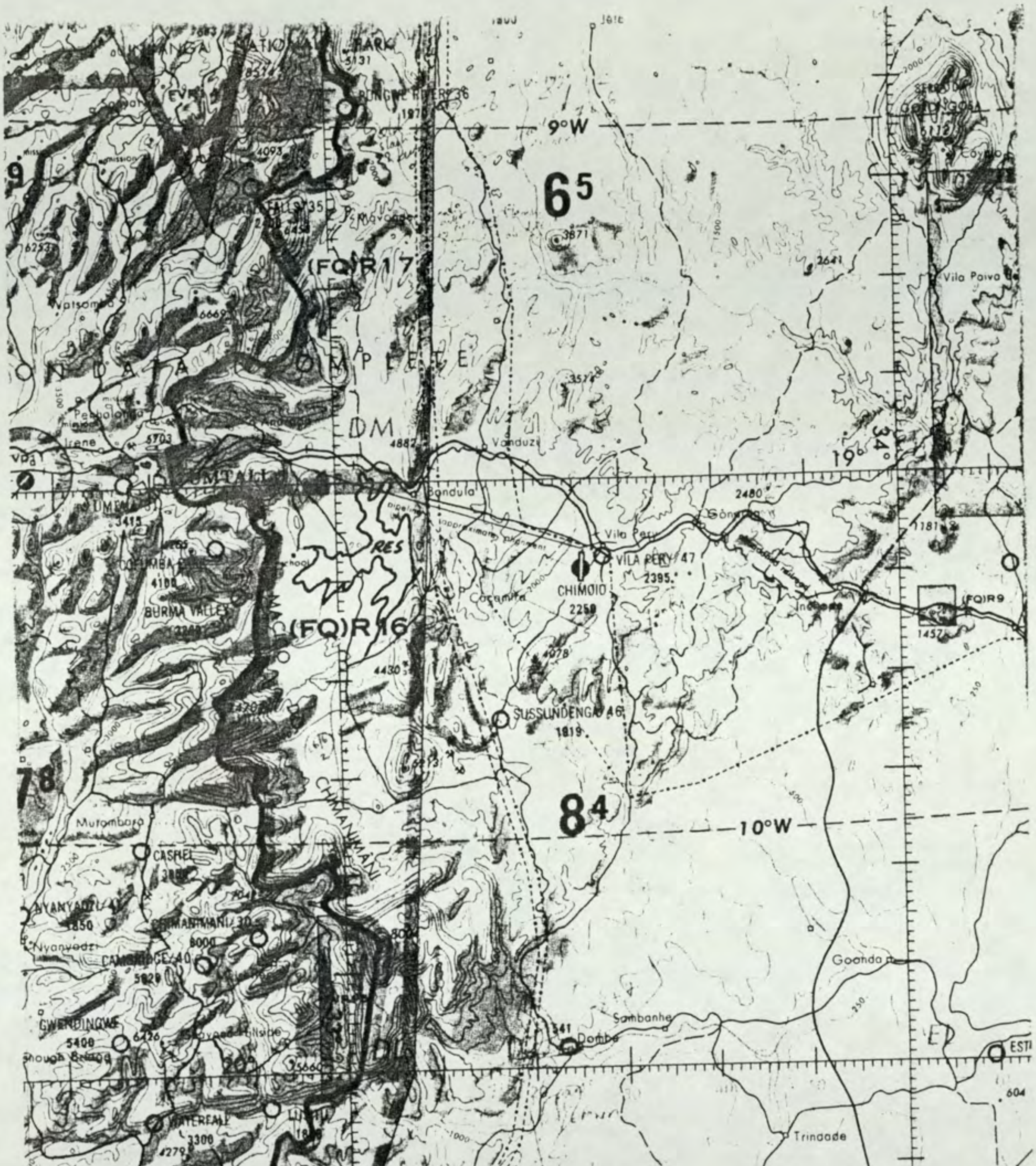
The positive images for 1986 were analysed, but in the absence of reliable ground truth, firm conclusions were not possible. The video-grabbed images were of such poor quality that processing was taken no further. The drum-scanned images were of reasonable quality and if a further band had been available a classification would have been made for comparison with that of the 1981 imagery. With band 7 and band 5 available it was decided a normalise vegetation index might prove informative. The result was disappointing with no obvious correlation between classes and the likely ground cover. It was impossible to investigate any correlation beyond a visual assessment without ground truth data. The main reason for the absence of ground truth data for 1986/87 was that field time in Rayong was limited and concentrated on farmer interviews. Cropping practice was recorded and photographs were taken at each interview site, but the majority of farmers had less than 8 hectares of land, with several different crops. Accurately identifying such small plots on the MSS imagery proved impossible.

5.7.2 Mozambique Case Study

Eleven classes were determined from the initial classification and all of these are shown on the examples which follow. Only nine classes were indicated on the classification for the whole province at a nominal 1:500,000 scale (given in the back pocket), as some clusters were combined after the field visit. Some of the classes correspond well to land cover divisions whilst others are very heterogeneous. Further processing of the data would require a thorough field survey. A description of each class determined follows, with some indication of the accuracy obtained. The accompanying example sets, with a full resolution extract of the initial classification, annotated with photographs of the actual scene, are provided as a visual explanation of the classification. Plates 5.14-5.18 appear on a separate examples sheet, with part of the initial classification shown for comparison (see back pocket). Most of the examples are from within the Beira corridor area as flying below 2,500 feet was relatively safe in places there. Figure 5.29 shows the corridor area at 1:1 million scale.

Since it was impossible to collect ground truth, the true accuracy of the classification cannot be presented by means of confusion matrices. What can be shown, is the accuracy of the classification algorithm in terms of the classification of test areas on each image, which looked the same as complementary training areas. A confusion matrix for each image, is provided on this basis in appendix F.

Figure 5.29 Showing the 'Corridor' section of the study area in Mozambique.



Plates 5.9 and 5.10 below are examples of Class1.

Plate 5.9 Showing indigenous hardwood forest classified as Class 1.

Plate 5.10 Showing a Eucalyptus plantation also classified as Class 1.

5.7.2.1 Forest and Plantation Cover

On the initial classification this was subdivided into classes 1, 2 and 3. The two photographs on the previous page show the diverse appearance both within the indigenous tree cover and a plantation. The plantation is at Sussundenga (see Plate 5.10) and the border of healthy eucalyptus shows as Class 1 (red on the classification examples), but the eucalyptus which appears to have lost it's leaves in the foreground of the photo appears to be Class 2 (brown) on the classification, similarly forest trees can fall into either class. Class 3 was actually a "shadow" class which could be partitioned between the two forest classes. A class for shadow was discriminated, to try and avoid confusion with water bodies and other dark areas.

5.7.2.2 Cultivated Land or Sparse Bush

An example of this category is labelled on plate 5.14 of the Manica examples set (see back pocket), where a very young (still grassy) pine plantation was classified as 4. Similarly, established orchards (where grass is grown between trees) were usually classified as 4. Unfortunately cultivated land, where maize and sorghum were still standing in the fields, also showed as class 4, as seen in plate 5.11. On closer inspection, the area labelled 4 appeared to be fields of sorghum or maize, still to be harvested.

5.7.2.3 Arable Land and Bare Ground

Examples of this class are shown on plates 5.16 and 5.17 of the example set and in plate 5.12. The dominant reflectance for this class is that from bare ground. Within the Beira corridor this is often land which is intensively cultivated and already tilled by June. Outside the corridor area this is usually bare, uncultivated and sometimes stony ground, including hill peaks. Thus it is likely to be infertile barren land over most of the province and yet within the corridor area, it probably represents some of the most favoured land.

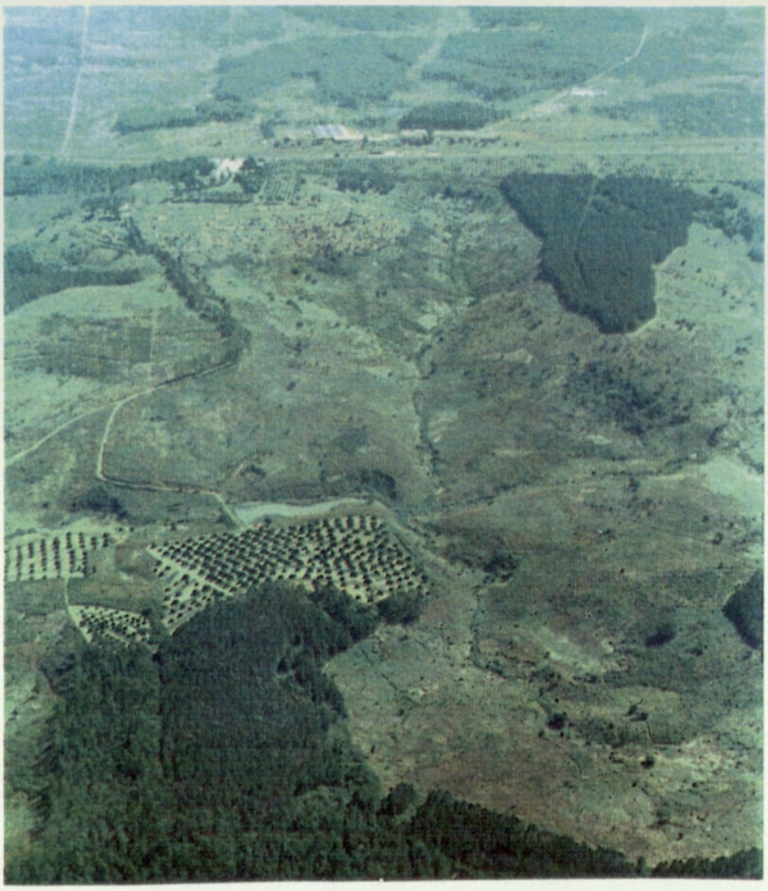


Plate 5.11 Showing how class 4 represented several different types of land cover.

5.7.2.4 Open Savanna

This category is shown in plates 5.12 and 5.13. It may occur as a natural climax type but from air observation it seems to often relate to land where bush is regenerating after slash and burn cultivation. It typically grades into the natural grassland (Class 7) and the dense Savanna (Class 8) occurring in the same areas.

5.7.2.5 Natural Grassland

This class has been named grassland, although shrubs and isolated trees may often be present in the land cover. The important factor in the discrimination of the class is the dominance of a distinct reddish hue from grasses present, distinguishable when flying over the area. This could actually be due to a grass of red appearance such as Themeda triandra or may simply reflect the state of parched grassland generally at this time of the year. Examples are shown in plates 5.12, 5.13 and 5.18.

5.7.2.6 Dense Savanna

The dense savanna category appears to have been accurately identified by the classification with little overlap into other classes. Examples of this type are shown in plates 5.11 and 5.12.

5.7.2.7 Moist Grassland

This category is indicated in plates 5.12 and 5.16 (back pocket). The instance in plate 5.12 indicates the very distinctive long thin bright yellow/green appearance of the dambos.

5.7.2.8 Intensive Farming

This category was accurately classified within the corridor area, where it was always found to correspond to intensively farmed areas, usually with irrigation practiced. Unfortunately, with the nature of the farming practice, which seemed to be to rotate the crops within the farm every year, it was difficult to determine the exact cause for the distinctive reflectance. Since this class also occurs in areas outside the corridor boundary, where intensive farming is unlikely to be practiced, it was suspected that bare ground was in some instances classified to this category. For the 1:500,000 provincial classification this category was grouped with class 5 to form a new class 6.

Plate 5.12 Showing the actual cover related to the classification (the key is given on page 187).

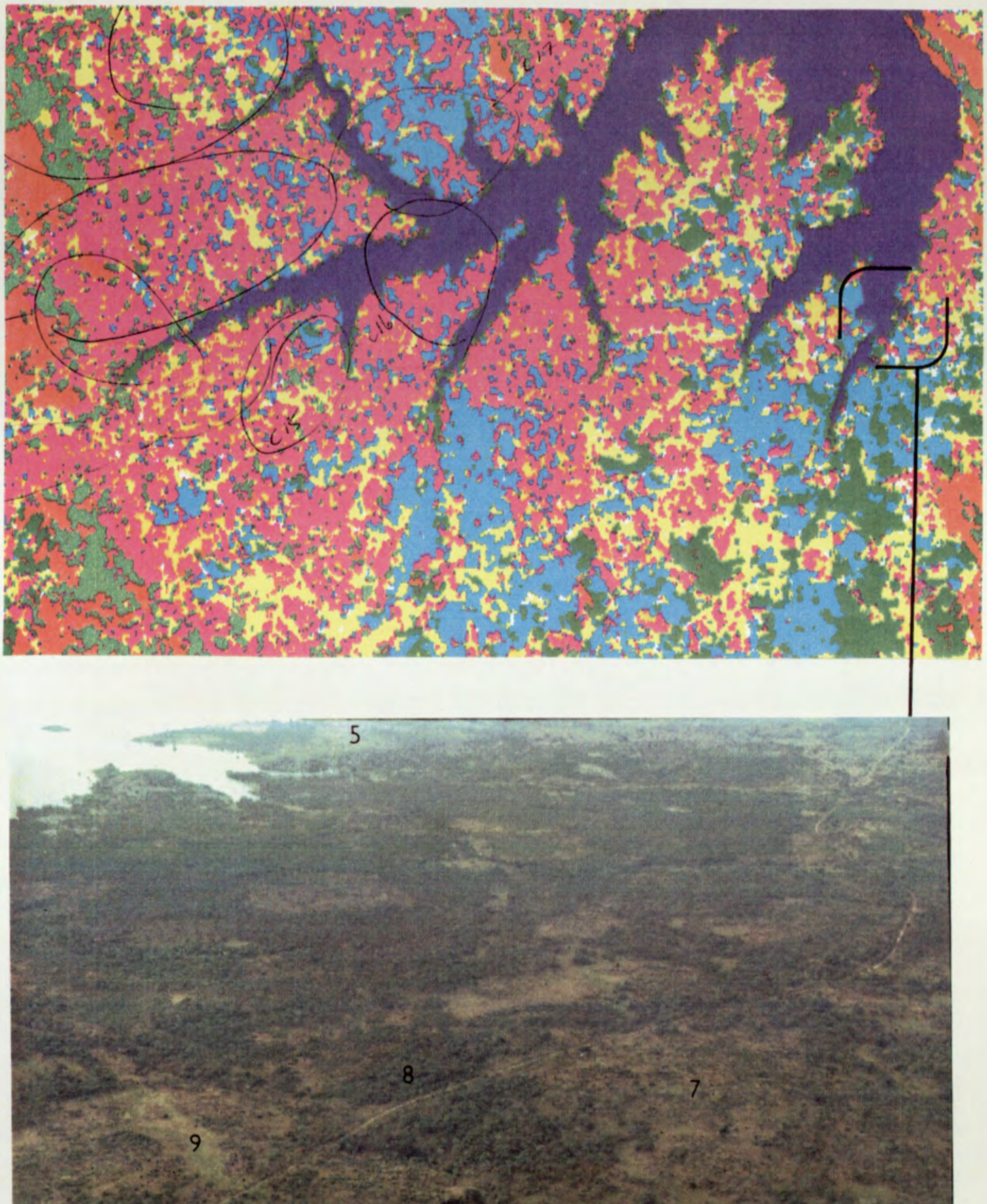
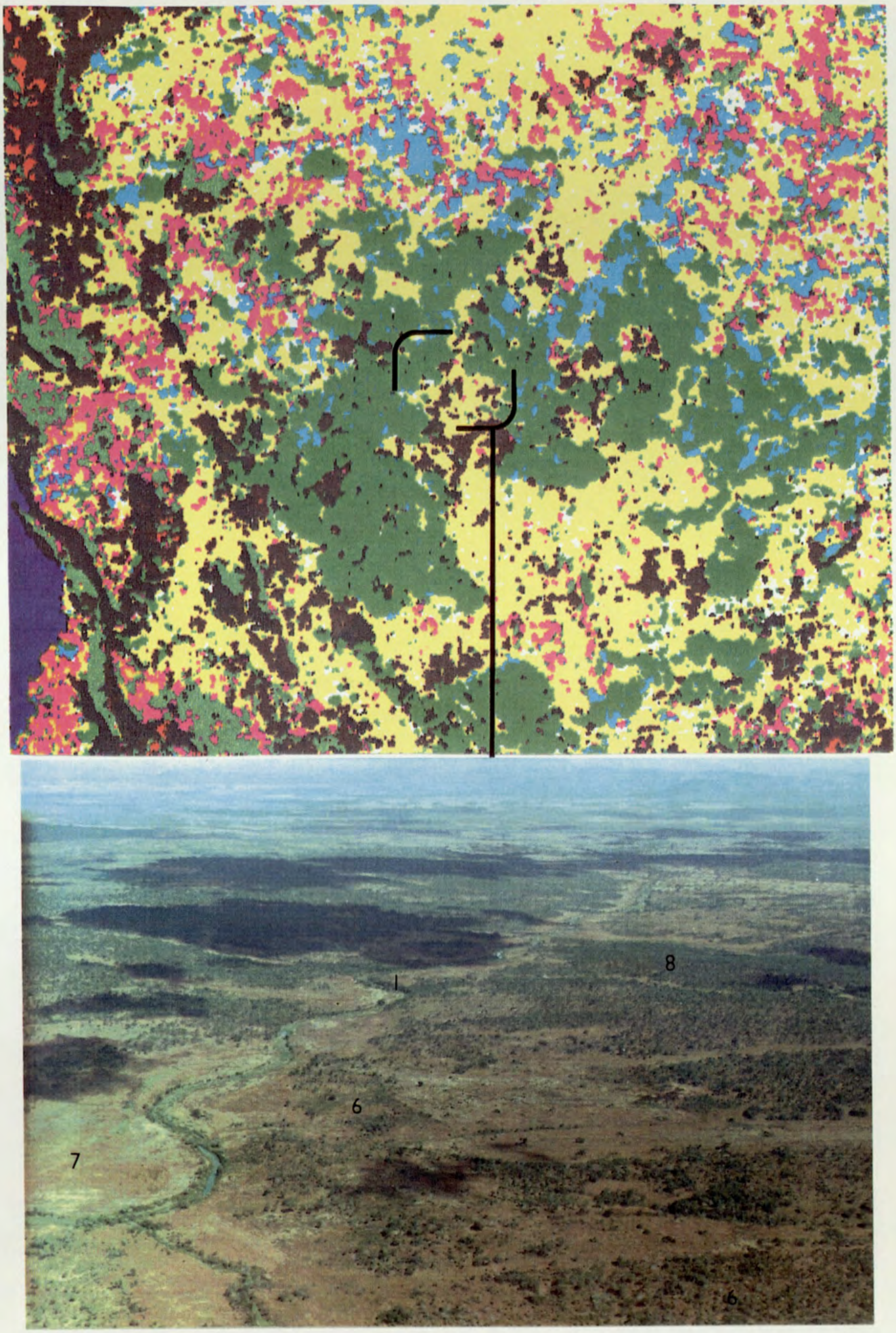


Plate 5.13 Showing the actual cover related to the classification (the key is on page 187)



Key to figures 5.12 and 5.13

Class 1 **Red** Plantation or Forest Cover

Predominantly plantation cover of Eucalyptus, but also sometimes pine or indigenous forest cover.

Class 2 **Brown or Orange** Plantation or Forest cover

Predominantly indigenous forest but also sometimes pine plantation and occasionally Eucalyptus.

Class 3 **Black** Shadow

Shadow caused by sharp relief, almost always corresponding to class 1 or 2.

Class 4 **Light Green** Farmland or Sparse Bush

Sparse bush cover; citrus plantations; Sorghum/maize; a very heterogeneous class.

Class 5 **Magenta** Arable land and Bare Ground

Bare ground which is often infertile rocky areas, but can also be recently cultivated land.

Class 6 **Light Yellow** Open Savanna

Secondary bush growth grading into classes 7 and 8.

Class 7 **Cyan** Natural Grassland

Grass/scrub cover showing a distinctive reddish hue.

Class 8 **Green** Dense Savanna

Dense savanna land with no evidence of previous cultivation.

Class 9 **Yellow** Moist Grassland-Dambos

Distinctive wet areas with lush grassland following drainage lines.

Class 10 **White** Intensive Farming

Corresponded to areas of intensive farming, often with irrigation in use.

Class 11 **Blue** Water

Corresponds to lakes and rivers where water was present in June, but most rivers were dry at this time and were delineated by tree lines or the white rocky river beds.

5.7.2.9 Water

On the central MSS scene this actually corresponded to water, but on most of the other scenes this represented dry watercourses or bare white rocks.

5.7.2.10 Inspection of the Hardcopy 1:250,000 Images

Hardcopy images covering the province for two different dates, November and June, were available. Discrimination of classes was totally different between the two dates. This indicates that multi-temporal data sets would probably significantly improve classification, but whether the improvement would justify the processing costs involved is problematic. There was generally less visible discrimination between cover types in the November imagery, so of the two dates, June was preferable. However, from discussion with national parks staff in Zimbabwe, it would seem likely that a slightly earlier date would improve discrimination of savanna types and this would also mean more crops were still in the fields. From inspection of the available data, it seems possible to obtain cloud free data in April for some years but this would mean a risk of interference from cloud cover, a compromise would be early May.

CHAPTER VI

IT TOOLS AND PROJECT CONSULTANCY

6.1 Introduction

This chapter considers the typical project environment, on an overseas land use planning consultancy job, in relation to the use of the IT techniques investigated. The aim was to investigate whether or not the IKBS and DIPS were likely to prove cost-effective, if implemented for a typical consultancy project, as encountered by ULG. Such a job does not usually extend beyond five years. For a provincial land use plan the total input of a land use planner would commonly be 12-18 man-months. Assistance in IT development might also be available from a systems analyst, say for six man-months. Many of the ULG projects are much shorter. Obviously project timings and technical assistance (TA) input, vary substantially with geographic location and the exact nature of the project. The above estimates are simply guidelines, to illustrate that the time constraints and pressure on subject experts, are such that there is very little chance for experimentation with new techniques. Thus in considering the possibility of introducing such decision support tools to the consultants armoury, emphasis must be placed on cost-effectiveness, as well as the final performance of the product. How fast a reasonable system can be developed, perhaps automating only 80% of the particular task, may be more important than how long it takes to achieve the perfect system.

The timing of system development can in most cases, be split up into component stages such as knowledge acquisition and knowledge representation, the use of satellite remote sensing is a special case. With satellite imagery, the physical data has already been collected, in the form of image files on a CCT. Such unprocessed data is however still not in a usable form. The processing of the CCT, to arrive at a thematic map of land use, can be regarded as paralleling the knowledge acquisition task, required in development of a knowledge based system. How the land use information is then integrated with other planning information, is a problem of knowledge representation.

6.2 Which Technique to use?

The detail of land use planning jobs undertaken by ULG in developing countries, has varied substantially, with each project having a different emphasis and varying in the baseline data available. Nevertheless, some aspects are common to nearly all such studies:

1/ A variety of primary spatial data sets must be compiled by survey or from existing studies, e.g. soil maps; climate; topography.

2/ Secondary spatial data sets must be derived from the basic resource data, according to specific needs, e.g. Erosion hazard maps; Available water capacity.

3/ Land use alternatives and their requirements must be considered, e.g. Agricultural crops; Forestry; Aquaculture.

4/ Environmental impact must be considered, e.g. effects on flora and fauna.

5/ Economic and Social analysis.

Most projects of this nature are still undertaken as manual studies, but this situation is rapidly changing, as consultancy companies and aid agencies alike, perceive the advantages of computer automation.

With so many spatial data sets (thematic maps) to compare and evaluate, most workers have immediately reached for a so called GIS, although as was explained in section 3.6, individual perceptions of what is a GIS vary. The majority of those implemented on research projects into land use planning problems, have been of the type relying on a relational database, linked to a mapping software package, as for instance in the case of the popular package ARC INFO (ESRI, 1984). Having a link from a database to a mapping system can prove invaluable, but it isn't actually essential for manipulation of the primary, or secondary spatial data sets. Once data has been pre-processed (usually involving normalisation to a standard attribute scale for instance S1,S2 etc.), data sets can be compared by logical operations on image files using an image processing system. From the work on the Rayong and Mozambique data, it was obvious that a spatial analysis system, suitable at least for provincial level studies, could be implemented on a micro-computer-based image processing system. The integration of satellite remote sensing analysis techniques with GIS, seems to have so far been undervalued. The major gain by using a DIPS for this stage of a project is in speed. ULG has in the past used a database system linked to a graphics package for comparison of spatial data and display/query, but it is both slow on querying and of coarse resolution. In addition, the input of data by means of an attribute value and geographic locator, for a given database field, is very tedious, error prone and time consuming. By use of DIPS techniques maps can be automatically digitised and stored as images. Comparisons are fast and results can be stored for almost instant retrieval. Criticism of such raster-based systems in the past , has centred on storage limitations, particularly for micro-computers, but this argument is no longer valid. The standard hard disk on a micro-computer purchased today is 40-80 Mb. As the following example from Thailand shows, such storage provision would be adequate for building a provincial land use planning system.

6.2.1 Planning at Provincial Level

Scales to be considered:- 1:100,000 up to 1:250,000
Highest level of detail 1:100,000

Primary Data sets to consider:

1) Soils series and association map

2) current vegetation

3) Topography

4) Forest boundaries (reserves etc.)

5) Special resource/constraint map (e.g. salinity)

6) Water resources/irrigation system map

7) Land unit map (derived from above and field survey)

The smallest area that can be drawn acceptably at the 1:100,000 scale is 100 rai in size i.e. 16 ha (160,000m^2), assuming a minimum dimension for polygons of approximately 0.4 x 0.4 cm as has been accepted by DLD.

A 512 x 512 pixel scene (262,144 pixels) typically requires a storage provision of 262,209 bytes (allowing for the header size). The size of area displayed would be 25.6 km x 25.6 km (655.36 km^2).

Surat Thani province is 12,444.47 km2 in area, this would require a minimum of 19 image files to cover the area. Allowing for about 25% overlap (or for irregularities) it can be assumed 25 images are required. This would typically require 6.575 Mb of storage for each provincial data set. For the 7 primary data sets 46.025 Mb of storage would be required.

Secondary or tertiary data can be generated from the primary data sets, so that storage requirements are not further inflated. Land unit information could in fact be defined as a secondary data set derived from base line information and so if economy on storage is required, then that data set need not be stored independently, but could be generated at run time. If it could be done by a look-up table operation it would be a very fast operation, with no significant time penalty attached to the indirect storage of information.

Digitisation of each map would take approximately 2 days, as compared with about 7 man-days for digitisation on a grid square overlay basis, for input to a database (timings based on actual project experience). The image based system would also perform faster on querying and would be able to display a higher quality image.

Typical output from the system would be a crop suitability map or a land use plan, derived by combining the results of a number of crop suitability analyses. To derive crop suitability ratings the requirements of a particular crop must be compared with the qualities of a particular land unit. To meet the requirements of a particular suitability rating values for different factors must be within a certain range for all factors considered relevant. This can be considered as a logical AND operation between factors, the result of each conjunction between operands being the lowest value in terms of suitability ranking. Such a function is easily performed between images on most image processing systems. Thus each factor would be represented by a bit mapped image and the final result would be an image indicating the suitability rating, for a particular crop on each unit.

Once the land unit classification has been made all the information required on land qualities for a particular unit could be accessed through this image and an accompanying database of land unit information. If a database is also held on the crop suitability information, then ratings could be determined for each land unit within the environment of a relational database, resulting in effect, in a look-up table of rating results, which can simply be assigned to a new image, on the basis of the land unit image.

This approach would save on the number of image files required and thus have considerable benefits in terms of minimising storage requirements. The database program may however lead to a reduction in speed of operation, as the code may be interpreted rather than running under a compiler (although compiled DBMS are now available). The advantages of the different approaches will depend largely on the sophistication of the hardware of a system, since image processing on some of the latest systems is extremely fast for simple routines, that are basically 'wired' in hardware, whilst some software may still run relatively slowly on the same machine.

Some information is not suitable for representation on an image. For instance, detailed information relating to an attribute, such as a land quality code, should be stored with the code label in a database. Also point source information often relating to the social conditions pertaining to individual farmers or communities, is best stored in a database. Whatever method is chosen for analysis of such data, be it an expert system, a spreadsheet program for financial data, or a customized program, it is vital that the database information and processed results can be related to the raster image files. This may require a link through a specialised programming language. Of the languages used in this study, both C and Prolog can be used to effect such a link, between a database and image files.

The use of satellite remote sensing is a special case involving the manipulation of spatial data. The difference between operations involving digitised maps and satellite imagery, is that the imagery is still in a raw uninterpreted form. consequently the processing involved, in arriving at a usable product, is more complex.

Knowledge-based interpretation of satellite imagery is an active research area and normalisation of digital thematic maps, could also be considered as a possible application for an expert system. However, from this work, the primary application identified for expert systems, has been for the incorporation of social influences. In the automation of land evaluation there is a great danger that computerisation will concentrate on the easily defined objective tasks, to the exclusion of important but subjective factors, relating to social perceptions and local tradition. Despite the problems encountered in using some of the expert system shells, the method of using a knowledge-based approach, to collect socio-economic information, as a means of modelling the local knowledge, was found to yield a reasonable computerised representation.

The detailed analysis of economic or financial information, is best separated from the expert system task in most situations, being better suited to a spreadsheet treatment. For the analysis of social information and perhaps even some physical data, knowledge-based systems are recommended.

6.3 Cost-Effectiveness of Knowledge-Based Solutions

The use of IKBS to automate a particular task, may yield benefits from cost saving, by a reduction in the time required for the task, or from an improved product or service. These two issues, of cost and quality are compared for the IKBS investigated in this study, as shown in tables 6.1 and 6.2 below.

Table 6.1 Showing the characteristics of the IKBS used.

	Handles	Can be	Score Out of 10	
	Large	Easily [1]	Ease of	Man-Machine
IKBS	Problems	Integrated	Querying	Interface
Micro Expert	No	No	5	3
SuperExpert	No	No	7	6
TIMM	No	No	8	7
dBASE	Yes	Yes	9	8
Prolog	Yes	Yes	9	10

[1] This factor relates to the systems as tested, improved versions in some cases are now available.

Table 6.2 Showing the size of the IKBS used and timings.

IKBS	Number of Rules Implemented	Development time (Man-hours)	Rules/Man-hour
Micro Expert	24	56	0.4
SuperExpert			
- Rayong	51	8	6.4
- Yasothon	158	16	9.9
TIMM	32	40	0.8
dBASE	24	200	0.1
Prolog	61	62	1.0

Table 6.1 shows that the dBASE and Prolog systems, both surpassed any of the expert system shells tested, in terms

of ease in use and flexibility in representation. The timings in table 6.2 indicate that the dBASE system requires a large amount of program effort relative to the return in terms of implemented rules. The TIMM and Prolog systems show a similar return to programming time, but the induction systems show a much higher return in terms of rules per man-hour. The figures given are indicative of the systems strengths and weaknesses but require careful interpretation.

For the comparatively small sized samples used, the rule return for the dBASE system is representative. This will change if larger systems are considered. As with Prolog, before rules could be added, a basic system to handle input/output and interpret the domain specific terms had to be built. It is noteworthy that there is such a significant difference between the rule return for similar Prolog and dBASE systems. Development in Prolog was much faster, the basic form of the system took about 3 m/days (or 21 man-hours) to develop. This suggests that with larger systems, the return would approach at least 1.5 rules per man-hour (m/hr). Using the results from section 5.6.3, that it took a total of five hours to add 13 new clauses to the existing system, this suggests a return of 2.6 rules per m/hr could be attained. Addition of rules to the dBASE system is trivial, so the return to programming effort will improve markedly as the system grows. Even if new attributes are required, these can be added without resort to programming.

The induction systems do seem to clearly out perform the other systems in terms of speed of rule development, but this is deceiving, because as was explained in section 7.3.2.2, such systems seek to derive as many rules as are required to discriminate the outcomes, even if these rules are spurious.

The data collection times indicated for each system represent actual time spent in the field, or collating information. Since the same Thailand survey data was used for three of the systems built, the collection period does not vary in those cases. There would be differences in data collection times between systems and it is instructive to compare these. This can be done from the experience established, with the various systems and field knowledge. The major difference is between the rule-based systems and the induction systems. In this context the dBASE system is considered as a rule-based IKBS.

From the work on induction systems, it was clear that many examples were needed, to hope to achieve realistic rules. For a provincial study, similar to the type conducted on Yasothon, the survey would typically be required to cover about 20 different types of cropping system. In Thailand, 10-12 major independent attributes were normally found to influence crop choice. Assuming 4 possible values per attribute and all attributes were important for each crop, then for each crop 40 examples would be required. In practice, not all factors are relevant to a crop and from the experience of the Rayong and Yasothon systems, the number of examples likely to be required is closer to 20. For 20 different crops this means a minimum of 400 examples are required, simply to train the system, without validation. Validation should be based on a similar number of examples. Because of the need to have representative examples, rather than random samples, the survey work must be closely monitored to achieve this end.

By confining questions to only those essential for the induction exercise, typical interview time would be 40 minutes and 8 interviews could be accomplished in one day. In Thailand such interviews would usually be conducted by two junior technical/extension staff working together. For 800 examples, on a per plot basis, a minimum of 200 m/days would be required to obtain the necessary information for a provincial survey. The information would still be in paper form.

If a rule-based approach was followed, as described in section 5.6.3 for Prolog, an expert would be required to elicit the information, plus a translator. Having completed the fieldwork in Rayong province, the author considered he had gained a reasonable knowledge of farming practice and was at the stage of being able to make sensible, if not expert recommendations. Thus the assumption of a requirement for about 50 interviews in a provincial survey, is reasonable for development of a basic system (which could be subsequently refined by local use). Based on the use of a portable computer with Prolog and development times given in section 5.6.3, an expert would require 24 m/days to incorporate 48 interviews, into an expert system, including design of input/output and any procedural programming required. To refine the system, the expert would normally wish to spend about 6 m/days gathering information from local specialists and institutions of importance. Validation would also be required, though this need not always be completed by the expert. Presuming the expert were to validate the system personally, a similar number of interviews as in the collection exercise would be required. Assuming the interviews were completed in half the time (since knowledge is only being checked, not elicited), this would require a further 4 m/days (6 interviews can normally be accomplished per day at the acquisition stage). This gives a total of 34 m/days to develop an expert system, representing the farmer's decision making on crop choice. By concentrating on elicitation of solely useful information, from the Thailand experience, an average of ten useful rules would be gleaned from each farmer. Thus an expert system of about 500 rules could be achieved in 35 m/days of expert time. A comparison of the timings for the two approaches is given in table 6.3.

Table 6.3 Contrasting development time between inductive systems and rule-based IKBS

	IKBS development time in m/days		
	SuperExpert	Prolog	
Staff required -	Junior technician (x2)	Expert	Translator
Activity			
Knowledge acquisition	200	12	12
Data input	16	Not required	
Programming	2	18	-
Validation	6	4	4
Technician time	224		16
Expert time	-	34	-
x2 for equivalence		68	
Total Time	224	84	

[1] Assuming an expert's time costs twice as much as local staff.

Rule-based expert system shells would be an easier alternative to Prolog, but a subject expert would still be required, so the timings can be considered a valid comparison, of the difference in effort required between rule-based as compared to inductive systems. It should also be emphasised that the end product of a rule-based system, is more flexible than the equivalent inductive system. The interface can be customized and integration with other programs can easily be achieved.

6.4 The Recommended Strategy

The unusual nature of overseas consultancy work in developing countries has been outlined. It is characterised by: High pressure and tight time constraints; yet the best possible results must be achieved for a company to remain competitive. Companies such as ULG have no choice but to adopt the latest technology, to maintain their competitive edge. The non-standard and knowledge dependent nature of the tasks involved, demands sophisticated solutions. This can make the choice of solution complicated and because of the often very narrow profit margin, particularly on short contracts, costly experimentation is not a viable option.

6.4.1 Spatial Data

Issues concerning the handling of spatial data were raised in the previous section. A very time consuming phase in a land use planning project, is the preparation of thematic maps, from other maps, field data, or remotely sensed imagery. This phase will have to be automated for a company to be competitive in bidding for contracts. Automation will also be essential to meet terms of reference, which are likely to be increasingly demanding, in terms of the information provision and flexibility required. This will happen naturally as the general awareness of what is possible improves. No doubt some countries and subject experts responsible for proposals will remain sceptical of IT techniques for some time, but these are likely to soon be the exception rather than the rule.

One option for automating the production of maps and general handling of spatial data, is the purchase of a GIS system, including the hardware required to run the system and for map production. Such a GIS would comprise a relational database and mapping software and until recently would only have been considered for use on a mini-computer. The situation has now changed and micro-computer versions are available. To produce high quality maps from digital data still requires a considerable investment in an appropriate plotter and other peripherals. If a project is specifically orientated towards establishment of a GIS and mapping facility, then the best technical solution can be offered and the price involved may be secondary. Unfortunately the situation is seldom so clear cut. Often the motivation for using an automated system for this phase in a study, is precisely because it will save time and yield a better product, but the equipment and software may not have been budgeted.

Under these circumstances the cost involved becomes very important. Another issue is that of portability. Where there is no specific remit to develop a mapping/GIS system, an expensive high quality system could only be used back at headquarters. Although this may be a sensible solution, unfortunately most clients will only consider paying for consultants time, whilst at the project location.

For the short consultancy jobs concerning land use planning but not involving establishment of a GIS as such, a Portable digital image processing system would be the ideal low cost solution. Several systems of this type are now available, most having quite sophisticated GIS software, as well as software for remote sensing analysis.

The main difference as compared to the higher cost options, is that the low cost systems usually run at lower speeds and the accompanying peripherals such as a colour printer, cannot provide as high a quality output. Neither is it usually practical to include a tape drive, reliance usually being placed on an optical disk. As was explained in section 6.2, an image-based GIS could be developed using only the image processing software, combined with links written in a language such as C. For many land use planning jobs such an image-based system would be sufficient, given the time constraints which usually preclude the possibility of building a full GIS.

6.4.2 Knowledge-Based Systems

The other stage of the land use planning process recommended for automation, was in the collection of social information, particularly concerning local expertise. Automation of this stage is regarded as especially important, to ensure such information does have an influence on the outcome of the plan. In considering how knowledge-based systems and particularly expert system shells might be used by a firm such as ULG, it is instructive to look at how others in industry have used them.

Two basic approaches to the introduction of expert systems in industry are recognised (Feigenbaum et al., 1988):

i) Prestige, large-scale ambitious projects to automate a major task completely.

ii) Dependence on subject experts, to launch there own small-scale expert systems after a short initial training and distribution of a simple shell.

An example of i) was the American Express (AMEX), where an expert system was written to automatically authorize charges (Feigenbaum et al., 1988). The vice president of transactions at AMEX believed that if expert systems were going to be useful to a firm, it was best to offer those of high impact and great value first. The budget available for the project was about one million U.S. dollars and it was to be completed in one year. Implementation was on a special-purpose LISP machine (an expensive computer specially suited to operating the LISP AI language). A supervisor of authorizations was chosen as the expert, to train the knowledge base and this involved the devotion of a total period in excess of 18 weeks over a year, together with regular communication to the team of knowledge engineers building the system. AMEX used a specialist knowledge engineering company on a contract basis, to build the authorizing system, involving a large team of knowledge engineers. The system was a success with estimated savings of $27 million per year and improvements in the consistency and effectiveness of decision making.

The second approach ii), to introduction of expert systems was favoured by the Du Pont company. Ed Mahler, the program manager of artificial intelligence at Du Pont, found that the primary issue in adoption and use of expert systems was not technology, but was ownership. The user must believe from the beginning that the system is his. Thus, Mahler states (Feigenbaum et al. 1988), the best situation is where the domain expert is the knowledge engineer, where he interviews himself, although this did not always prove possible for Du Pont. At Du Pont expert systems act as assistants doing 80% of the work of experts. Achieving the remaining 20 % by an expert system would not be cost effective. Mahler found that employing outside firms to solve knowledge-based problems for Du Pont, was not effective. He believed the reason was, that although external experts were well equipped with knowledge engineering skills, they don't know specific jargon and are mistrusted by highly trained technologists and scientist. Arrogance of the average knowledge engineer was also cited as a problem.

A typical expert system at Du Pont requires a man-month of effort to build and saves $100,000 per year. 'Payoffs' recognised were mostly from the replication of expertise, but also improvement in quality and consistency in decision making. Mahler reported a return on software and labour costs, across the Du Pont group (the seventh largest corporation in U.S.), of 1500% and an aggregate saving of $10 million for 1987 (Op. cit.,1988).

The only feasible approach for the ULG style of operation, would seem to be the latter (ii), whereby subject experts act if interested, because:

1) Experts have very little time available to help in developing the knowledge base, as fee-earning work takes precedence.

2) Experts are seldom at base and may have to travel at short notice.

3) An expert system application would only be budgeted as part of a particular job. Persuading a client of the worth of such an exercise is very difficult, unless he is already converted to the idea of an expert system and the contract was bid for, on the basis of inclusion of such a system.

4) Because all expert system development work has to be done on the basis of fee earning contracts it is not possible to assemble a team of knowledge engineers, for the sole purpose of such development. Such experts time must be chargeable to a particular job.

By using the subject expert to develop a knowledge base, the problems of long interview sessions and constant interaction between a knowledge engineer and subject specialist are avoided.

A consequence of this approach, is that the AI tools to be used must be easy to operate. In other words expert system shells would be preferred, over the building of customized expert systems in an AI language. Some subject experts may have adequate computer experience to build their own systems based on the precepts of IKBS, but in the language of their choice. As computer literate professionals become more commonplace in industry, more esoteric systems may be developed, but in the meantime trusting to expert system shells, seems the most appropriate recourse. Until recently (1989) many shells were woefully inadequate for implementation of useful systems, including most of those tested here, but this situation has changed. Some of the essential provisions of an expert system for land evaluation work as identified in this study can now be found in some of the latest shells available. The crucial requirements for a shell or programming environment to be used in land use planning are listed in section 7.8. If software meeting these requirements can be provided to subject experts for their experimentation, new applications are likely to quickly become evident. A training course would be given in the use of such decision support tools and backup provided on technical problems with the software. Beyond such support it would be up to individual experts to decide where and when use of such tools was appropriate, if at all. The only direct expense to the company would then be the software and training course. If viable prototypes were subsequently identified, further funding might be considered, provided a saving or improvement in service could be shown.

CHAPTER VII

DISCUSSION AND CONCLUSION

7.1 Micro Expert

For the application tested with Micro Expert, it was assumed the knowledge domain was completely understood, so that the problem was simply one of knowledge representation in an efficient, straightforward and user-friendly manner. Thus the comparison was between using this expert system shell and an alternative means of programming the information, for instance with a conventional language like C.

One of the principal objectives of an expert system shell, is that the non-programmer should be able to build his own system (see section 3.3.2). This objective has not been met satisfactorily in Micro Expert, a substantial effort is required to understand the syntax of the 'Advice language' almost akin to learning a new programming language.

A second problem was that although a tree structure is meant to simplify control of a problem, the way probability was propagated through that structure was not easily understood. The reliance of the whole system on a representation dependent on probability meant that in modelling a simple problem with little uncertainty in the database, the problem seemed to become unnecessarily complicated and programming bugs were not easily corrected. In a useful system of larger size, refinement and maintenance of the program would be a burdensome task for the subject expert.

Another criticism is that although the 'Advice language' is not easily learnt it does not actually support a wide range of commands. The example of the Bayesian rule, given in the results section, contrived to provide categorical output, was the only way found of supplying to the user, an answer in a reasonable form. As can be seen from the session records given in appendix A the man-machine interface was very primitive, and would probably leave a novice user who hadn't built the system, rather non-plussed. This would especially be the case where the problem was not normally considered in terms of probability, but the answers necessarily have to be provided in probability form.

Micro Expert was not found to provide a very 'user-friendly' interface. Even if this had been the case, the very limited provision for mathematical or statistical operations, would necessitate a facility to pass data in and out of other programs. As was stated in the results, the link to PASCAL programs could not be made to work with the version of Micro Expert used. The ability to switch in and out of the expert system during run-time is likely to be crucial to building a useful classification system and would certainly be essential in the case of modelling resource management systems.

7.2 Knowledge Acquisition by Questionnaire and Informal Interview

7.2.1 Knowledge Acquisition and the Farmer

One of the particular aims of the field work and analysis for this study was to establish a suitable method of knowledge elicitation for 'capturing' farmer experience. This was with the further aim of arriving at sensible rules to be used in deriving a land use plan, at provincial level. As was explained in section 1.2, it has been traditional to ignore the experience of the farmer, but as many would argue (Richards, 1985), before deciding on a land use plan the people involved should be consulted first and their knowledge and experience should be tapped, to help in planning their own development strategy. **The question to be answered was whether IT tools could assist in incorporating the procedure of knowledge acquisition into the planning process.** This was investigated by three main methods:

a) automation of data entry by utilisation of a DBMS;

b) automatic induction;

c) and the use of Prolog for fast-prototyping.

The data entry and handling using dBASE was more a tool to aid in processing than for actual data acquisition, but in the sense that loss of important data through slack data handling, is an example of poor knowledge acquisition, the dBASE system will also be considered in this context. Automatic induction and Prolog represented two alternative methods for quickly capturing farmer expertise and the two methods were compared and contrasted in section 6.3. Essential to any of the IT methods to be discussed was the interview, and the questionnaire design. The following section is concerned with a discussion of the results from the interviews (a summary of questionnaire results is given in appendix G).

7.2.2 Farmer Interviews

The three field visits made to the Eastern province of Rayong resulted in a wealth of information on the local farming systems, collected by interviewing local farmers. The objective of the interview was to identify important factors considered by the farmers in choosing which crop to grow. The intention was to categorize the values of such attributes, according to the farmers view of how such an instance (set of attribute values), affects crop choice. An understanding of how farmers arrive at their decision to grow a particular crop, was essential to building a system to predict the choice of crop a farmer would make. This was to be compared with an evaluation based on the FAO style system as implemented at that time by DLD (see appendix G).

A decision had to be made as to whether an informal but structured interview was the best approach for knowledge elicitation, or whether a pre-set questionnaire was more appropriate. To avoid pre-judging which factors were important to farmers, an informal interview was favoured. Against this had to be set the fact, that much of the information required could be predicted and was required as background anyway, or needed for the FAO style classification. In addition the DLD officers accompanying the author were not very experienced in informal interview technique. The adoption of a fixed format of questions, was considered the most reliable approach for collecting consistent data. To accommodate the requirement of flexibility, certain questions (Numbers 5,6 and 7, as in appendix G)were designed to allow the farmer to comment. These were :

1/ Why has the farmer chosen to grow each of his crops?

2/ Were there any crops the farmer no longer grows?
If so, why?

3/ For any crops expected but not grown, why is the
farmer not growing them?

These questions were designed to identify additional factors and important limits not covered in the fixed questions. The answers to these questions, together with detailed notes made on discussions with each farmer, were used to arrive at the rules for the Prolog system (see appendix E). The induction systems used the answers to pre-set questions (including some like slope and capital which had to be appended), in conjunction with the list of crops grown, to try and induce a decision tree for crop selection. This method was aimed at covering all possible factors accepting that some information might prove redundant.

Information on financial details such as crop price and costs was also collected, in the expectation that financial considerations would be very important. This indeed proved to be the case as was evident in farmer replies, such as:

'I grow durian because it makes me the best profit'.

Unfortunately it proved very difficult to use the financial information in any of the systems built. It was used in the dBASE system to indicate the likely return for a particular crop, but it was never used in determining which crop to grow, which is how a farmer uses such information. A possibility would be to use an index based on the return to one rai for each of the crops under consideration, given average conditions, this might then be built in at a low level of detail in an induction system. With the shell systems used there would not be enough available attributes to allow considerations of the financial return for each crop at the same level as physical factors. In this sense it is not possible to actually emulate the farmers reasoning. Such a constraint does not apply to the declarative prolog system, although for site specific calculation of returns, integration with a mathematically orientated language would be required.

It was obvious from the field work and the preliminary attempts to use the data collected, that emulation of farmer reasoning was not a trivial task. The major problem identified was in gathering sufficient information. This was a twofold problem: potentially a lot of information can be used by the farmer; and eliciting such information is a time consuming process. The first problem is that of the large number of factors which may be assessed in arriving at a decision. Often farmers only concentrate on a few principal factors such as the net return, which was the most common answer given.

The situation is complicated by reliance on a few complex factors, such as 'the market' or the 'type of land'. A farmer would commonly say 'this is not durian land', such an affirmation might be made because of depth to water table; soil texture; topographic position; slope; or a combination of these and other factors. To emulate the farmer's simplistic description requires gathering information on all the detailed factors concerned. Additionally, each farmer tends to consider different factors important. This widens the set of factors to be considered in a pre-set survey. To cover all factors requires a lengthy and detailed questionnaire, which leads to the second problem, that of accomplishing a survey in reasonable time.

The Rayong questionnaire was revised and 'pruned' to the minimum of questions felt to be required, but despite this it took on average at least forty minutes to administer. Farmers were generally very cooperative, but asking for forty minutes of somebody's working day is an imposition and in some countries and situations, this could be a more significant problem. The necessity for translation complicated discussion and great care was needed to avoid misinterpretation of questions. Obtaining accurate information on labour requirements and yield figures was usually difficult, as most farmers could only make a rough estimate. Overall the results from the Rayong surveys were informative and comprehensive. The major problem was that instead of completing an anticipated 150-200 questionnaires only just over 50 were completed. This was because the length of interview required had not been accurately anticipated and the journey time between selected farms was considerable. Despite this, it was felt a clear picture had been gained of the farming systems in Rayong, the information had been gleaned but not necessarily in the required form.

It was anticipated that more individual instances would be needed for the induction systems, but this was impossible to achieve within the time and resources available. Fortunately the information from a survey on Yasothon province conducted by DLD, provided a test case for the induction systems with many more examples. The problems with that survey data, symptomatic of a project with ill-defined objectives, are discussed in section 7.3.2.2. Information was only collected on a few of the important attributes, there was evidence that the questionnaire was too long. The declarati'e rule-based system written in Prolog was not directly affected by the number of examples collected, as it was only dependent on the quality of data. The significance of this is discussed in section 7.6.2.

7.3 Inductive Expert System Shells

7.3.1 Expert Ease

The rule formed in the crop choice system for Brunei was given in section 5.3.1 of the results. The following sections comment on the first rule formed on the raw data.

7.3.1.1 The Domain

By considering those attributes on which class limits have been imposed, two hundred and forty different combinations were given in the domain. By including integer attributes with no limits there is a very large combinatorial explosion. The fewer the values an attribute can assume, the less likelihood there is of an outcome being associated with a particular value by affects this. The domain of possible values should be considered and any obvious groupings of values should be made. This can be an interactive process by which a rule is refined with successive categorisation on the basis of decision points detected.

7.3.1.2 The Training Set

The knowledge engineer must ensure there are adequate examples of each attribute value in the training set. Each possible outcome must be given for each value. If a certain value is never found with a particular outcome, it will be assumed the land use is not feasible, if that value is present. So omission of examples is very important, and should signify a positive step. If knowledge of the effect of a particular attribute is lacking then the wild-card symbol could be used, to indicate that as far as is known, any value is permitted with a given outcome.

7.3.1.3 Class Limits

The decision tree showed the attribute value categories selected by the system, as important divisions in determining land use. This in effect represents what the farmer perceives as important limits. For instance it was suggested that the slope limit for growing rubber in the farmer's view may be about 29^{o}. The perceived sensitivity of horticultural crops to waterlogging, was also indicated in the use of water table to separate fruit trees and horticultural crops on sandy loam soils. With comparatively few examples the limit detected of 70 cm is probably a very rough guide, but with more examples limits might be revised.

7.3.1.4 Dominant Attribute

Soil texture was found to be the attribute yielding the most information on current land use. According to the classification, it would appear in many cases that land use could have been determined solely on the basis of texture. Although valid in the case of peat swamp, this was clearly false for such a texture as sandy clay loam (SCL). The anomaly has been caused because with fifteen different texture values included, the example set of forty one sites is inadequate. In the only two examples of SCL soils, fruit trees were growing in both case. Because the system had no examples of other crops growing on this soil, it was assumed SCL soils always indicate fruit trees.

7.3.1.5 Generalisation

The first inclination might be to dismiss such a classification as of no use, but in fact the classification has operated correctly on the information available. There are two possible options for improving the inferred rule. One is to increase the number of examples to cover more combinations of attributes. The other is to reduce the complexity of the problem by generalising the soil texture classes in such a way that one class is then covered by more examples, without actually expanding the example set. Unless hypothetical examples were to be introduced there was insufficient data to expand the example set, so textural classes were consolidated from fifteen down to four.

After categorisation of the integer attributes, the anomaly of decisions being made simply on texture alone was removed. Instead of a flat form the decision tree shown in figure 5.3 is much more hierarchical, as more attributes have been used to determine the land use. This is because by categorising the data there is more chance that a given outcome will have the same attribute value as another outcome. This makes the classification more realistic and meaningful, although there is obviously a danger of over-generalisation leading to more clashes, where different outcomes occur for the same set of attribute values.

Such generalisation will tend to increase the number of attributes needed to define the outcomes. This is shown in the clash which occurs in the rule, between the Rubber and Dipterocarp forest outcomes. Such a clash indicates that differentiation between the two outcomes, is inadequate in the example (training) set. Either more attributes must be introduced, or the number of categories within certain attributes should be increased (reducing generalisation). Of course for the land evaluation problem it may well be that at least in terms of physical suitability, certain crops are equally suitable. This would simply mean that crop choice was decided ultimately on economic/financial, or social factors.

7.3.1.6 Applications

Expert Ease was the first of the inductive expert system shells tried. For this particular problem the example set was found to be too small, but no further examples were available. The application of induction was successful, in the sense that information on farmers judgments of the land was summarised in the form of a rule. The rule was more informative than a collection of soil survey sheets (see the example sheet given in appendix B), and the rule could be interrogated.

The system was very easy to use, no previous computer programming experience was needed. It was possible to arrive at an informative and meaningful (although not completely accurate) classification very quickly. The major task was the input of data. There was no means of reading in data from another file. This was irrelevant to the Brunei exercise, but would be important under a project situation where time is of the essence, and much data may already have been digitised.

The ID3 algorithm detects which attributes are of importance, so that all possible attributes can be included and any that are irrelevant will not be used. The rule also indicates through the clashes, at which points further data is needed for more detailed differentiation between choices.

Text annotation for the final system was easily achieved, such that a subsequent user could query the rule, though there is no provision for adding contextual help and the screen display format was limited. For example, with a long list of choices for a particular attribute, the choices have to be displayed on two screens and even if the choice is known, a selection cannot be made until the second screen is displayed. Options cannot be displayed width-wise across the screen as was done in the dBASE system (see appendix D).

The same approach could be applied to large sets of results from field surveys of crop performance, linking recorded yield to chosen parameters. Another possibility would be to use research station results of cropping trials, although in normal circumstances such data would have already been thoroughly analysed and would usually only be considering one crop factor.

The search of the ULG archive of land use planning projects, revealed that on many jobs there would simply be insufficient data, for an inductive system to arrive at any results. This could signify one of two things, either judgements on cropping practice have been made on the basis of internationally accepted guidelines, with a transfer of recommendations, known to be appropriate in similar circumstances elsewhere; or local knowledge was already available in an exploitable form. The local knowledge may have been documentation on local crop requirements and yield figures, or may have been through the advice of local experts. The latter is much more likely. If there is a reliance on the use of local expertise, as suspected, then this should influence the type of knowledge based approach to adopt.

The results of this initial test of an inductive system on a desk exercise, were sufficiently encouraging to proceed with the improved version of the system (SuperExpert), in a field situation. The intention was to see whether in practice inductive shells could enhance the analysis of field surveys of farming expertise.

7.3.2 SuperExpert

In SuperExpert the man-machine interface was an improvement on that achieved with ExpertEase but, the query screens still relied on a multiple choice selection by pressing a number key. The facility to introduce examples prepared externally and presented in a standard ASCII file proved to be very important, as it allowed problems of a realistic size to be investigated. This would also allow the SuperExpert shell to be incorporated with other software, in a land use planning system. Although problems were still encountered in running SuperExpert off a hard disk.

7.3.2.1 Rayong Province

The super expert system for Rayong Province was designed to build a decision rule by induction, from the answers given by farmers, to questions covering what were thought to be key attributes. In designing the questionnaire forms (see appendix G) the following attributes were identified as being likely to be of importance: Tenure; capital; transport costs; dependence on other employment; other business interests; long term outlook; management expertise; labour supply; size of farm; depth to water table; whether the land floods; and the soil type.

The first rule formed from the Rayong data (Rayong 1 in appendix B) illustrates clearly some of the problems inherent in application of induction systems, only three factors were used in the rule. If a non-expert assumed the rule to be true, some very strange recommendations would be given. For example, the rule suggests that:

'If the farmer has previous experience in growing rice and the soil is an association of Ry 48 and Ry 46 then either vegetables, cassava or rubber should be grown.'

This is a nonsensical rule and of no use. The three crops named, may well be grown on similar land, but the socio-economic circumstances influencing their choice differ significantly. The problem lies in the large number of possible values (17) for soil type. False relationships may be assumed simply because there are not enough examples of each soil type. Yet some of the soils have very similar properties. With 17 possible values and only 60 examples it is unlikely there will be examples of every possible crop, for each soil type. It would be unusual for only one crop to be associated with one soil type.

The soils were grouped according to the parent material and physiography and this gave a rule that involved more factors. Despite this, tenure; Experience; Alternative employment; Availability of family labour; Local wage rate; Availability of oxen; Availability of a tractor or power tiller; Depth to water; ownership of water pump; and loans were all factors unused. Only a few factors were used : Moisture regime; Soil series grouping; Slope; Age ; Area; Capital and Other Experience, 7 out of a possible 20 factors. Yet there were - clashes which suggests the factors were not adequate to describe the situation. With more examples the factors may have been adequate, since with an expanded data set the choice of order in testing factors can change and this may cause other factors to be checked. This is a weakness of the ID3 algorithm in that as soon as the outcomes can be divided on the basis of certain attributes no more attributes are tested. Unfortunately this means that once committed to a certain hierarchical representation a new tree cannot be superimposed to investigate the unexplained clashes. Figure 7.1 shows the hierarchical tree relating to part of the Rayong 2 rule.

```
Rubber : MOIST
    F : SLOPE
        B : RUBBER
        ABD : NULL
            AB : SERIES
                ALLUVIAL : CLASH-DURIAN-RUBBER-COCONUT-RAMBUTAN-MANGOSTEEN
                    EROSALLUV : NULL
                        EROSION : NULL
                            PIEDMONT : CLASH-DURIAN-MANGO-RAMBUTAN
                                UNDULATING : NULL
```

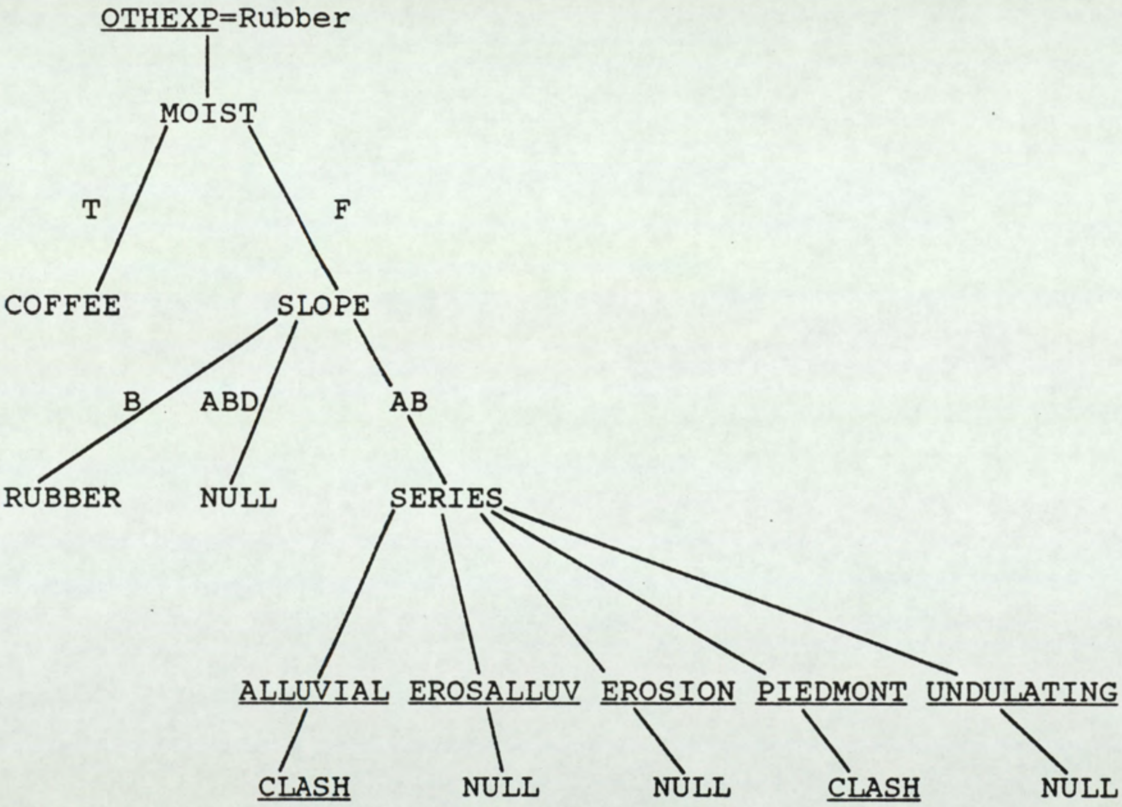


Figure 7.1 Showing the hierarchical decision tree imposed as a fixed rule to describe the Rayong data.

From personal experience of farmer interviews, income from alternative employment; availability of family labour; and depth to water table, were all important factors for many farmers, in deciding on the crops to be grown. Tenure was also very important, but in Rayong it only effects a very few farmers since most farmers are owner occupiers.

Income from alternative employment was found to be important, because such a reliance often dissuaded farmers from growing more labour intensive crops. An example was Noy Kareewong of Ban Nog Haen, who had another job as a truck driver. He planted para-rubber, partly because he could then avoid the concentrated effort required with cassava, at certain times of the year. He said that durian required too much management (i.e. many different time-consuming operations, like spraying throughout the year), and that he had insufficient labour to grow vegetables. The problem of insufficient family labour was exacerbated by his alternative employment.

The depth to water table was an important factor in Rayong province, because durian which was a very profitable crop, requires a lot of water. If the water table is at great depth it is not feasible to dig a well and pump water for irrigation. Many farmers were not growing durian because they believed they did not have sufficient water. For instance, Boon Song of Khao Vai, was growing para rubber, because he felt his land was too dry and water supply too unreliable for durian. A similar problem exists for coffee which also requires regular irrigation.

The information on farming systems, collected by personal observation and informal interview, as well as the pre-set questionnaire, shows that the induced rule was not an adequate representation of the decision making process.

7.3.2.2 Yasothon Province

As was explained in section 4.4.2.3 of the methodology, without detailed knowledge of the farming system in Yasothon province and no opportunity for field work, it was not possible to test stringently the validity of the rule derived by the induction process, but with the data set of 617 records it was possible to draw some conclusions about the methodology. The aim was to test whether a rule could be derived to determine what farmers were likely to grow. This included examining the methodology for collecting the examples and comparing the efficiency and success of this method of knowledge acquisition, with other methods attempted (see section 7.6.2 on Prolog methods).

The collection of the data used in this expert system was undertaken with the main aim of providing general socio-economic information for the Economic Section of the Land Use Planning Division of DLD. It was possible to include questions of relevance to an induction system because assistance was given in design of an appropriate questionnaire and programming of a data entry system in dBASE III. This highlighted one of the problems in the inductive approach, that of acquiring the data. Obtaining sufficient examples for a comprehensive system requires a considerable effort in terms of manpower and resources, for the fieldwork in Rayong this amounted to 50 m/days and for Yasothon the estimated effort was 150 m/days. Unless such information is to be collected as part of a wider survey, a consultant for a particular job, would normally be required to undertake and organise this himself, the cost and time involved would be considerable (as detailed in table 6.3, section 6.3). In this case the opportunity was available to collect the information as part of another survey, but several problems were encountered. These problems stemmed from the major disadvantage that a general survey was adopted, designed by a committee of in excess of twenty people, leading to a very lengthy questionnaire which would take over two hours to conduct thoroughly. From the data collected there is evidence that questions were skipped, particularly those requiring comment. Questions were not skipped consistently though. so that selection of a complete attribute set was very difficult, this reduces the number of attributes, which can be tested. The questionnaire was administered by people who were not involved in the ultimate processing of the data and this may also have had some bearing on the care taken in completing forms, but this is inevitable in such a large scale questionnaire. A shorter tailored questionnaire would have resulted in a larger and more uniform data set, but this would not normally be available, as it is unlikely to be useful to others.

The other consideration in handling the basic data, was preparing the data in a form that could be accommodated by the SuperExpert system. Only integer and logical values are acceptable. Some of the numeric questionnaire information was stored in dBASE as character information, this had to be converted to integer types and categorical data represented by numbers, had to be altered to begin with a letter.

The rule formed by induction given in appendix B does not show any obvious anomalies in terms of ridiculous outcomes, but beyond the 617 record domain, omissions cannot be detected. The hierarchical rule does not compare with the evaluation principle in the FAO framework, of testing every attribute of importance. This inductive rule can be seen as a stage preceding a final evaluation, whereby important factors in considering a particular crop are identified. For instance, the rule suggests that in deciding whether or not glutinous rice and groundnuts in rotation, is a suitable option, only the land unit and age of the farmer need be ascertained. This may indeed be the case if the land unit was such that the physical resources were so limiting as to restrict options down to a very few choices.

The rule also shows how attributes are not seen as being independent, which is probably more realistic than adoption of the FAO ideal, of independent land qualities. It is very difficult to maintain such independence when introducing socio-economic factors. For instance on some land units representative of flooded conditions, questions on irrigation or tenure are likely to be irrelevant, as the former would be inappropriate and the latter probably incidental. Despite these comments some of the rule nodes suggest that the example set was still inadequate. According to the rule, in land unit Y4 the only option is Mali rice, no other attributes need be tested. This would be very unusual and demands further investigation. It may be explained by an unusual constraint such as salinity which is a problem in Yasothon. It may of course simply mean that very few questionnaires were administered for land unit Y4 and so a biased sample was taken. In the case of land units Y6 and Y13 this resulted in no examples for these values. In selecting examples for the training set as many examples as possible were taken for every crop, but it is difficult to arrange that there are also adequate examples of every attribute value. This must really be achieved at the interview stage, but requires careful monitoring. Obviously there will tend to be a bias towards more information on the land units occupying the greatest area, unless careful stratification is adopted. To ensure best possible use of the examples is made in the training set, close inspection of the data set would seem to be required, but this conflicts with the objective of automating the decision making process and may introduce unpredictable bias.

The results indicated that a still larger data set was required. An attempt was made to use all the Yasothon data, gathered in a single training set, even though this would leave no questionnaires for validation of the new rule, but with the system used an error was given after 420 examples had been loaded and the system crashed. Although no limit of the system was apparently broken, the problem seemed to be linked to reliance on a floppy disk for the system files, so that space was not adequate for the problem. The Yasothon 289 example system occupied 168 Kb without the query text, which is a very large file for a comparatively small problem. This is another difficulty since it indicates the approach is not very efficient. On overseas work the computer hardware available is often out of date and large memory requirements for small problems would certainly be a constraint to implementation.

The clashes which occurred were not inconsistent with what might be expected, as for instance in the case of different mango types. This indicates more detailed information would be needed to decide between varieties. Sometimes a clash indicates an unrealistic division imposed by the system, in an effort to separate outcomes at all cost. This is shown in the code excerpt and hierarchical tree shown in figure 7.2.

In reality all the attempted thresholding of integer values shown in the code excerpt, is inappropriate and there should only be one outcome viz. glutinous rice/mali rice. The expert can achieve this by renaming of the crop choices and categorisation of integer attributes into logical attributes. The approach then becomes an iterative process to build a sensible system. Ideally, all possible factors likely to be involved in crop selection, should have been included in the system to begin with and redundant factors could subsequently have been excluded. This is not practicable for two reasons; First the system is only designed to accommodate 20 attributes; secondly the problem already highlighted arises, of how to collect such comprehensive information.

```
 < 52 : AGE
    < 49 : DOYRS
      < 31 : AGE
        < 43 : GLUT
        >= 43 : CLASH GLUT - MALIRICE
      >= 31 : GLUT
    >=49 : MALIRICE
>= 52 : GLUT
```

Or in tree form:

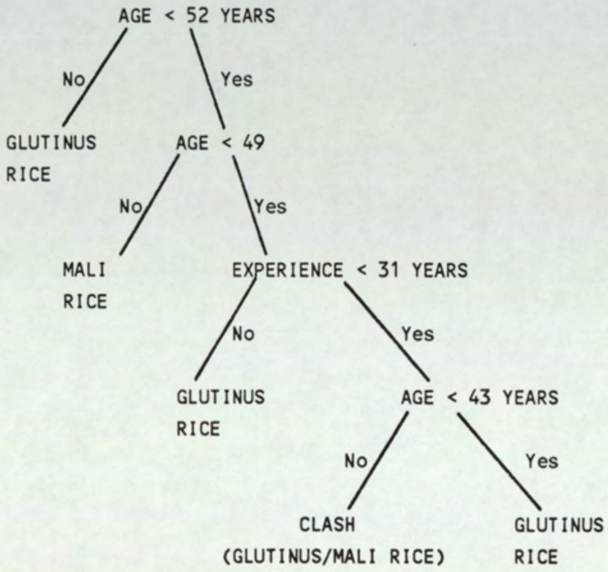


Figure 7.2 Showing how dubious thresholding can occur, as the system seeks to avoid clashes.

7.3.2.3 The Factors Relevant to Induction

Table 7.1 Major factors of importance to farmers in deciding their choice of crop.

Factor	Available from Questionnaire (Yes or No)	
	Rayong	Yasothon
1 Land Unit	N(Soil series)	Y
2 Local Crop Price	Y	Y
3 Water Availability	Y	Y
4 Tenure	Y	Y
5 Farm Size	Y	Y
6 Capital	Y	N
7 Labour Availability	Y	N
8 Experience of Mechanisation	Y	N
9 Marketing Opportunity	Y	N
10 Alternative Income	Y	Y
11 Current Crop	Y	Y

Table 7.1 shows factors considered to be of major importance to farmers in deciding which crop to grow (decided from personal experience in Thailand). Also indicated are the factors available from the Yasothon and Rayong questionnaires. In the wider context of an overall development plan rather than individual farmer choice, many more factors must be considered, Richards (1985) lists the socio-economic factors of particular importance, as given in table 7.2.

Table 7.2 Important factors, in assessing local skills and possibilities for agricultural development (After Richards, 1985).

Productive Activities

1. What does the village produce? e.g. farm crops; gathered produce; from manufacturing and processing or by services.
2. Who produces what? i.e. are certain jobs specific to men, or women, or another sector.
3. Access to productive resources? i.e. what resources are needed for production and is access to those resources restricted by society.
4. What happens to the product? i.e. is production for subsistence, local markets or international markets.

Local Skill

1. Description of the main farming systems with emphasis on local practice and crops .
2. Note local vegetation and soil-management skills
3. Local use of livestock. Who owns animals and how are they managed?
4. The importance of fishing and hunting , i.e. Unusual activities or reliance of a particular sector of the community on this activity.
5. Procedures for selecting plant materials or culling animals. i.e. is selection consciously practiced.
6. Local food processing technologies.

Facilities, Equipment and raw materials

1. Transport. Describe procedures and costs involved in local transport.
2. Tools, machinery and household equipment.
3. Water supply. List the main water sources.
4. Raw materials. Source, costs and trends in supply.

Social and Political Organization

1. Organization of labour i.e. Pattern of labour inputs and identification of seasonal bottlenecks.
2. Political organization, i.e. Local political and administrative system organization, who would need to be involved in an agricultural development project.
3. Local perceptions of development priorities.

Richards contends that all these factors (he also lists questions on manufacturing and construction skills), are important in arriving at sensible development strategies for the rural sector. This is also the author's own experience, but the task of assembling such information is very considerable, in terms of time and effort. The factors listed in table 7.1 should summarize all the more detailed and pertinent facts, gathered from a survey of the type urged by Richards. Unfortunately this is very difficult to achieve in practice by a single visit with a questionnaire. What actually happened, was that particular factors such as depth to water table or ownership of oxen were used to represent 'water availability' and 'mechanisation' respectively. The fact that this did not work very successfully, is shown in the neglect of most socio-economic factors in the rule formed. Water availability is a complex factor in Rayong related to the water table depth, rainfall reliability, capital available, the size of plot to be irrigated and labour availability. Most of these factors were included independently for Rayong but the indication is that the methods of measurement, or detail provided, were inadequate, as known relationships were not detected by induction.

FAO (1983) list the following predominantly physical factors (over the page) as land qualities to be considered in designing a land evaluation scheme. The factors listed are not ranked, they can be considered as independent and of equal importance. In certain instances true independence may not hold, as for example with a land unit characterised by flooding where lack of water is unlikely to be a problem (although this is possible in the dry season). Despite this water availability is such an important factor that it must be separated from the land unit description.

FAO listed Land Qualities:

1. Radiation regime
2. Temperature regime
3. Moisture availability
4. Drainage conditions
5. Nutrient availability
6. Nutrient retention capacity
7. Rooting conditions
8. Conditions affecting germination
9. Air humidity
10. Conditions for ripening
11. Flood hazard
12. Climatic hazard
13. Excess of salts
14. Soil toxicities
15. Pests and diseases
16. Soil workability
17. Potential for mechanisation
18. Conditions for land preparation
19. Conditions for storage and processing
20. Conditions affecting timing of production
21. Access within the production unit
22. Size of potential management units
23. Location
24. Erosion hazard
25. Soil degradation hazard

In the Yasothon data, the land unit is a complex factor encompassing a diverse set, of consistent land quality values, including some of those listed above (see appendix I for land qualities used by DLD). The other factors listed are socio-economic parameters not usually considered in a typical FAO style land evaluation, except perhaps indirectly through the management level. Even water availability can be included as an economic factor, as it is often a question of whether the farmer has sufficient funds to dig a well and purchase a pump, yet it is obviously equally a physical factor, as in many cases water may be readily available.

The current crop grown is a particularly important factor, since in a province like Rayong, where plantation crops prevail, the farmer is unlikely to make a hasty decision to go out of say rubber or durian, for short term reasons. Also, some crops, are commonly grown in rotation or as inter-crops (see appendix F for crop list). Thus crop choice is not a totally independent consideration. It is actually more realistic to recognize common crop mixes and use these as outcomes. Questions on the secondary crop need only then be asked once the best choice of major crop had been decided. The inter-crop would then be decided at a lower level in the hierarchical system and would probably be more sensitive to current market conditions and subsidies.

Thus it can be seen that the provision of a successful land use plan, can only be achieved after a very detailed analysis of the current situation. To collect all the information needed, requires a considerable amount of fieldwork beyond a simple questionnaire survey. Land evaluation can be considered at two levels

a) Based on the actual `rules of thumb' and socio-economic pressures as experienced by the farmer.

b) Based on physical rules derived scientifically.

This kind of separation is sometimes acknowledged but then level a) is usually ignored because it can be very difficult to model. Farmers often apply physical rules as well so that the concept of different levels is artificial, but it is a useful distinction to make in considering questionnaire studies. At level a), rules would normally only be applicable to a small area. Therefore for a queationnaire study to be of any value two key rules should be observed.

a) Data collection must be properly stratified.

b) Objectives should be limited and well defined.

The approach of asking many questions in a study over a large area is not likely to be of any value in most circumstances.

Questionnaire surveys should concentrate on collecting simple socio-economic details and cannot be expected to provide sufficiently comprehensive and consistent data for the inductive systems investigated. The inductive systems implemented were only designed to emulate the farmers reasoning, but as table 7.2 shows, the preparation of a comprehensive land use plan, involves many more complex factors than those considered. It can be concluded that automatic induction should only be the first and even minor phase, in an exercise to give the preliminary guidance on important attributes and thresholds.

7.4 TIMM

The system to predict farmer's decisions built in TIMM worked satisfactorily to an extent, but as the exercise session showed, the success rate of decisions judged to be correct was poor. Where the example tested was very similar to those held in the knowledge base, the decision given was very sensible, but as with the induction systems there were too few examples used in training, resulting in an inadequate knowledge base. For comparison with the induction systems the knowledge base was deliberately left undeveloped, since there were no more examples available. The knowledge base could have been improved by adding rules based on the author's experience, but this would have invalidated the performance comparison with the induction systems. By the same token hypothetical examples can be invented, to train the induction systems until more accurate predictions are attained, but this is not then a test of the knowledge acquisition and representation capability of such systems. When rules are given from experience, the expert is acting as the instrument for knowledge elicitation rather than the IT software.

There is a major difference between the induction systems and TIMM in the type of inference engine used. As explained in section 4.5.1 of the methodology, rather than induction TIMM uses a method of analogical matching based on statistical analysis. The rules closest in attribute values to the current instance under investigation are identified and crops recommended are selected from those rules, with a certainty attached according to the combination of certainties from each rule used (see section 4.5.3). The method used by TIMM is well suited to handling the circumstance of having several suitable crops. The certainty for each crop could be used to indicate the approximate proportion of farm acreage to be devoted to that crop, although that might be better decided by a subsequent analysis using a linear program. With SuperExpert a crop combination was only shown as a clash, so that without investigating the rule a user could not even check which crops were concerned. Also, there was no means of knowing which crops should be given more weight, no certainty was expressed in the induction systems.

The partial match algorithm of TIMM reduces the reliance on examples compared to induction systems, because it is not necessary to have an exact match in the rule set. This has two effects: There is always an outcome, it is not possible to be given a 'null' response; and a partial match gives all possible crops, rather than a clash which is the equivalent outcome if categories are widened in SuperExpert. An exact match in TIMM is equivalent to a unique outcome in SuperExpert, except that TIMM has the advantage of offering a crop combination as the outcome. This can only be done with SuperExpert if the system is trained on crop combinations as the outcome, rather than individual crops.

The other advantage of TIMM is that unless the 'narrow down' option is chosen, all factors will be considered, unlike the induction systems. This means there is less likelihood of false assumptions through an inadequate data set as can occur with SuperExpert.

The inability to link a TIMM system into another application program would be a severe constraint to using TIMM to build a land use information system. There is no graphics capability, no adequate provision of mathematical functions and the text editor was very poor. As such TIMM could not be recommended as suitable for LIS applications. In principle all importan+ TIMM functions can be accessed via calls to FORTRAN subroutines. This should facilitate it's incorporation into user-friendly evaluation systems. The version tested could not be made to work properly in subroutine mode. Since later versions will probably overcome the linking problems, TIMM may well be considered for a limited role in a LIS. The representation of farmer's knowledge was found to be adequate, although somewhat laborious to achieve. The size of the system and inadequate problem parameter capacity (just 15 choices), would however mitigate against successful implementation.

7.5 dBASE

7.5.1 The Intention

The dBASE DBMS was evaluated for the purpose of addressing the aim outlined in sections 1.1, and 1.3, of examining the application of knowledge-based programming to land evaluation. The application of dBASE to the problem of automating land evaluation, was set in the context of previous attempts at automation, found to be inflexible; demanding of programming expertise; expensive and difficult to integrate with other systems, as explained in chapter II. The intention was to try and improve upon the previous attempts at automation, by using dBASE.

7.5.2 The Low Cost Solution

The dBASE III software used runs on a 16 bit IBM compatible micro computer and retails at about £300, which is considerably cheaper than most dedicated software packages and particularly some of the expert system shells being promoted. Thus it certainly fulfills the criteria of a low cost solution, even after accounting for programming time (see section 6.3).

7.5.3 A Knowledge Based Approach

One of the chief problems experienced with land evaluation systems written in a conventional programming language, is that dependable portability is seldom achieved. In a new situation, programming expertise is usually required to alter the system, to accommodate new crops or different limits. Examples of this situation were given in chapter II, the LECS and FCC systems both showing failings in this regard. The reasons for the lack of flexibility in other systems are not always self evident, as such a failing is not simply a function of the programming language used, but rather a question of the programming methodology adopted. From this study it has become evident that it is possible to program in a knowledge-based style, without using accepted knowledge based software. Knowledge based style is defined here, as applying a clear separation of knowledge and data and a declarative rather than procedural approach. The knowledge based programmer should address the tasks identified by Simons (1985), as given in section 3.2, which can be summarised as: knowledge structure; Encoding; Control; Flexibility and Knowledge elicitation. The dBASE III evaluation system written, followed the principles of knowledge based programming.

The knowledge structure of the problem was closely dependent upon the data structure of the DBMS, which was based on the relation data model, as described in section 3.1.2. The knowledge base was conceived as a set of rules in the form of tuples for the crop relation. The crop requirement details formed the knowledge to be applied and these were held as records in the CROPSUIT.DBF. So each record (tuple) could be thought of as a rule in the form:

"if 'DRAIN'=X and 'SLOPE'=Y and ----- THEN CROP is Z"

The analogy with the structure of rule based expert systems like TIMM can be clearly seen. The system was designed to access information on land units already entered in a database (LANDINF.DBF), but it could readily be altered to allow query for land units not previously encountered. In either case the data would be held in the LANDINF.DBF file with partially the same relation structure as that in the CROPSUIT.DBF. So although the structure of the knowledge and data was identical they were clearly separated.

The inference engine (see section 3.3.2.1) determining how the knowledge was to be used, was principally implemented as the matching procedure, described in section 5.5.2.3. Pattern matching between the land unit information and possible combinations of attribute values, for different crop ratings, could have been achieved in a simpler way, if there had only been a very limited set of possible combinations describing crop suitability. The two databases could have been queried together with a 'JOIN' command, to form a new file storing the crop and rating for the crop requirement record which matched. The condition for the join, would have been for the conjunction of all the attribute values for the land unit, to be found in the crop record. The problem with that approach is that any one of the attributes can be limiting and cause the rating to be downgraded. Each attribute typically has three or four different category values and twelve attributes were checked. This would yield a possible 16,777,216 combinations. Many of these would be ridiculous combinations, but nevertheless the fact that only one of 12 attributes may be the limiting factor, makes the consideration of all possible tuples in the relation unviable. It was for this reason that the supposed independence of factors and the rule of finding the single most limiting factor, was turned to advantage in designing the matcher procedure.

The matcher procedure interrogates the value of each quality in turn, matching qualities between the land unit information and crop suitability knowledge, independently of other qualities. The 'AT' construct was used to achieve this. This allowed several values to be stored in the same record and attribute, of crop requirement information, as was illustrated in figure 5.15. This was necessary because several quality values may be graded equally. The 'AT' function checked for the occurrence of the land unit value for each attribute, within a list of values in CROPSUIT.DBF. The matcher procedure is completely independent of the number and names of factors involved. The factor information is accessed from the FACTOR.DBF which can be updated without programming knowledge. Similarly alterations to the knowledge base of crop information through the 'OPTIONS' menu had no affect on the matching algorithm. The independence of the matching algorithm from the data and knowledge ensures flexibility. In a new situation with different crop requirements, the user would simply update the crop requirement knowledge from within the system and run an evaluation, no programming would be required. The emphasis would be placed on collecting local knowledge, rather than following assumptions on cropping practice transferred from other countries.

The only procedural emphasis was in the programming of the 'EVALUATE' and 'CHANGES' programs, to impose a control structure on the search mechanism and to handle input and output. The actual data access in searching for a match at runtime is non-procedural. A query is declared and a match is then sought for each quality and for all relevant crops. When querying for all crops the search starts at the top of the 'CROPSUIT' database and finishes at the bottom. This is very similar to the exhaustive search of all clauses, for a matching pattern, in the Prolog program (see section 5.6.2).

Knowledge elicitation was not one of the knowledge based programming tasks addressed by the dBASE system, no attempt was made to incorporate automatic knowledge elicitation. As was stated in section 5.5.1 the crop requirement knowledge used was based largely on the existing DLD guidelines and recommendations from the agricultural compendium (De Geus, 1973). It would be possible to use dBASE III in knowledge elicitation at the stage of querying farmers and local experts. This would only be viable if a pre-determined question format was applicable. In such a case, an entry screen would be written into a program, to store the information to a database file. The dBASE program would then replace a paper questionnaire. The advantage over a paper questionnaire would be that transcription at a later stage is avoided. Thus the chances of losing data through misplacing documents or mis-interpreting answers is reduced. The use of a database with a fixed structure, also ensures questions are answered in full according to the required format. The disadvantages are: that it is often difficult to be so sure of the structure required in advance; if technical problems arise with the computer, the end result will be a mixture of written answers and digital data; and portable computers are still often considered a costly luxury on a project, so that they may not be available.

The speed of the dBASE program was disappointing in comparison to compiled languages and the TIMM expert system, but this would not necessarily be an important constraint. The slow speed is due to the nature of interpreted languages whereby the source code is assembled into executable code at every iteration. Searching is also time consuming and the time obviously increases with the size of the database. One refinement which would improve performance would be to alter the matching procedure so as to stop checking quality values as soon as the rating was registered as S4, since further checks would be irrelevant. The latest version of dBase can be compiled so speed will be substantially enhanced.

Relational databases provide a reliable means of modelling a land evaluation system, superior to procedural languages such as FORTRAN, because they provide the flexibility to adjust the knowledge base. They do not generally provide any noticeable advantages in overcoming knowledge elicitation problems beyond streamlining of initial processing. As with any other computer based land evaluation system, the results will only be as good as the data provided. By forcing the users to supply their own knowledge base, results are likely to be improved, but this may lead to a bottleneck at the knowledge acquisition stage. It proved easy to incorporate other language routines with dBASE although only by means of common data files, rather than by direct procedural calls. If a series of different tasks are to be performed on the same data, by different programs then this approach would be satisfactory. In the case of a LIS however, the perceived need was for one overall program depending on calls to whatever language provides the best means of modelling each sub-task. For the general automation of land use planning, several independent programs may be the most sensible approach (built by subject specialists) and under such a policy, a relational DBMS would be very useful for coordinating the results and providing a common database.

7.6 Prolog

7.6.1 Representation

As with a rule based expert system like Micro Expert or TIMM the means for representation of knowledge are rules of the 'If---Then' form. The difference lies in the level of constraint on rule structure. Rules in TIMM require all relevant factors to be identified prior to programming (as is also the case with Micro Expert) and although the types of format allowed provide adequate means of representation, the poor text editor results in a limited variety in question layout. The inductive systems were used to derive the rules themselves, but equally this required some prior knowledge, of the attributes likely to be most important. The expression of attributes was constrained to logical values or integer types and it was not easy to provide elaborate query screens, at the enquiry stage. The dBASE system designed also required prior knowledge of important attributes and the rules applied could not be entered in near natural language, but had to be programmed in the dBASE command language. As was shown in section 5.6.1, in Prolog rules can be entered virtually as they would be expressed in natural language. Without further format alterations they can be added to a Prolog program and take affect. In some cases names may need to be changed and other subsidiary rules added later, as described in section 5.6.3.

7.6.2 Program Structure

One of the major characteristics of the Prolog language is that it is declarative with no enforced procedure from the ordering of different clauses. The logical construction of the knowledge base of rules dictates the order in which different goals are resolved to solve the ultimate goal, of in this case crop selection. The consequences of this declarative structure are:

a) Normally only relevant questions are asked of the user.

b) The programmer does not have to make explicit procedural connections between factors (rules), to arrive at a rule solution.

This is in contrast to a system like Micro Expert, where dependence on probability meant great care had to be exercised in any alteration to the rule base. In Prolog new rules can easily be added without affecting the integrity of the program.

7.6.3 Fast Prototyping

By avoiding the need to consider procedural aspects from the inception of programming, development time with Prolog did appear to be significantly reduced. The respective programming times for each system designed were given in table 6.2 in section 6.3. Although not directly comparable as each system differed slightly in it's intent and the degree to which it was developed, the results show that development time in Prolog, was significantly less than by use of rule-based expert systems or DBMS.

Programming time is only one factor in developing a working system. The integration of programming and knowledge acquisition, through the use of Prolog, is discussed in more detail in section 7.6.5.

7.6.4 Problem size and complexity

No problems were detected in knowledge representation with Prolog. This is in contrast with the Micro Expert system where the syntax of the `advice language' was found to be limiting. Prolog was capable of handling all data types encountered and the graphics support within the language allows the man-machine interface to be customized as required to provide an attractive and easy to use interface. Unlike the expert systems tested, Prolog is not constrained to simple problems with limitations on the numbers of attributes and values. The only limitation to the system designed would be available computer memory. Even the memory constraint can be avoided by use of predicates which can access database information from disk files.

7.6.5 Knowledge Acquisition Using Prolog

A Prolog system of the kind demonstrated with **FARMTHAI.PRO** could be programmed at the very inception of the knowledge acquisition process. Programming could begin with the act of interviewing a farmer. If a portable computer was available, key points could be recorded immediately in Prolog in the form of rules. In most programming languages there is only limited advantage to the immediate input of data, as it must be subsequently transformed into a useful form, in which case the computer record serves much as a notepad would. The advantage of Prolog is that since the data (knowledge) is the program, the recorded facts can be `run' with very little additional programming and no change to the virgin data. This has several important ramifications.

a) By using this method the problem of losing or misconstruing information through transcription is eliminated.

b) The knowledge gained from interviews is available immediately to be used while the problems are still fresh in the mind of the specialist. When collecting data by questionnaire or through notes, material is often left unanalysed for some time, through the inertia involved in processing the questionnaires into a usable form.

c) By using an expert to collect the data personally less interviews are required because the expert knows what information to concentrate on and can identify which interviews will be useful.

d) Iterative training of the system can begin at an early stage, perhaps on the same field visit, thereby saving expensive time in the field. There are often difficulties of access and cooperation on field visits. By continuing the development of a system on a portable computer, until sensible results are obtained, the use of field survey time would be optimised. Often where repeated field visits are required lengthy gaps between visits, due to administrative problems, conflict with the objectives of the fieldwork.

e) The four previous points made, all illustrate improvements to the knowledge acquisition process, but the end result yields an improvement in accomplishing the task of developing a useful predictive system. The approach would lead to an early completion of a prototype computer system to predict the most appropriate choice of crop. This could then be provided to local experts to test evaluate, and improve, at a much earlier stage than would usually be possible in a project.

7.6.6 Emphasis on Constraints?

The problem with the above approach is that care must be taken in grouping factors and until a large number of interviews have been completed the correct associations will not be obvious. This adds to development time and devalues some of the advantage of instant recording.

An alternative might seem to be to emphasise constraints.

This would mean listing all criteria in a negative manner so that each factor would be tested to assert that there were no constraints for that factor. With all factors tested the crop could be safely recommended. This would mirror the process used in the dBASE system for land evaluation discussed earlier. The difference is that the factors important to farmers are not independent land qualities and this is typical of socio-economic factors, their importance tends to vary with individual circumstance. Whether an attribute is considered a requirement or a constraint it's effect will depend upon the state of other factors. The grouping of factors, as illustrated in the previous section for the cassava clauses, could be regarded as a process for identifying socio-economic qualities, similar to the concept of a land quality. The detection of dependent relationships between socio-economic factors, is likely to be even more difficult than successful definition of land qualities, as important relationships will change dramatically with location and circumstance.

7.7 Satellite Remote Sensing

7.7.1 Rayong Case Study

7.7.1.1 The Spectral Classification

The minimum distance classification of Landsat MSS imagery from December 1981, for an extract covering Rayong province, was found to have an overall accuracy of 60%. Training areas for the classification relied on a contemporary land use map from the DLD. The accuracy was also established using other areas from this map. Pineapple, cassava and paddy rice areas all appeared to be accurately identified at over 70% accuracy. There was about a 1 in 2 chance of the rubber or coconut classification being correct and areas of sugar cane were not accurately identified (only 15%). The results are a reflection of the cropping practice found in Rayong, but two factors connected with the image processing, affected the accuracies reported in the confusion matrix:

i) In choosing training and test areas from the DLD land use map only large homogeneous areas were selected.

ii) No threshold was applied to the classification to create an additional class of 'other' pixels, not classified as one of the identified crop types.

It was sensible to choose large homogeneous areas as training sites, but the choice of similar areas for testing the classification, was in hindsight likely to overestimate the accuracy of the classification. Ideally the test areas should have been pixels selected at random, whereas the test areas were selected because they could be easily identified. On Landsat MSS imagery identification of site areas can be a problem, so location of randomly selected pixels would be very difficult. The land use map from DLD was very detailed, but location of small sites on the imagery was extremely difficult as features such as rough tracks and small streams cannot usually be identified. For the area of paddy rice, the boundary from the DLD map was actually digitised and geo-referenced, to the same coordinates as the imagery. This meant every pixel could be compared and this did indeed give very different estimates of accuracy, compared to the 94% indicated in the confusion matrix. This would have been impractical for every land use type as digitisation was quite time consuming. The confusion matrix can still be considered to give a reasonable estimate of the accuracy with which large crop areas were identified.

The second factor affecting accuracy was the threshold level. The number of different spectral classes to be investigated was decided by visual inspection of the imagery, discrete homogeneous areas were compared with the

DLD land use map to identify possible classes of interest. The six classes named in the results were the only categories to be positively identified. It was known that a much more varied agricultural pattern existed, but this was not obvious from the MSS imagery. To allow for the other crops and crop mixes which could not be clearly identified, another miscellaneous category could have been added to the classification, by imposing a threshold. With a minimum distance classification this would mean ensuring that a pixel would not only have to be closest to a candidate class, but would also need to be within a prescribed distance of that class. The distance threshold is often specified according to a number of standard deviations from the class mean. Such an approach would not have indicated anything about the nature of the ground cover in the miscellaneous class, but could have improved the estimate of the area of crops identified. A problem arises in deciding the precise threshold level to use. Thresholding was tried initially but because of the extent of overlap between classes it had little effect and the final classification was presented without thresholding. Inevitably this means some of the classes incorporate other crop types, leading to errors of commission. Thus although crop areas may be quite accurately identified, as in the case of cassava, other areas not growing cassava are also included in the estimate (in the case of cassava the commission error was 42%).

The classification showed that with Landsat MSS imagery, it was possible to estimate the areas of some of the crops grown on a large scale in Rayong to reasonable accuracy. This was true of pineapple, cassava and paddy rice. All these crops are also grown on small scale plots, but the majority of the production in the case of pineapple and cassava comes from fairly large farms. Rubber is also typically grown in quite large plantations (usually about 8 ha.), but the accuracy of identifying those areas seems to have been poor. There is a lot of fruit tree production in Rayong and it is likely that most of this was classified with the rubber category, although as with rubber one of the problems is the varied appearance of plantations, until they are fully established. Young rubber or fruit trees may be intercropped with pineapples; cassava; maize; cassava or groundnuts between the rows, or the ground may simply be left bare. In contrast the large farms of cassava or pineapple, particularly the latter, have a fairly uniform appearance for most of the year. It was surprising to find coconut was not more accurately identified (50% omission and 42% commission), as this crop is usually very distinct on other remotely sensed images such as air photographs. Identification on air photographs relies primarily on texture though, so the spectral distinction may not necessarily be as good. The addition of texture measures to the minimum distance classification would be likely to improve the spectral classification of coconut and probably other crops of distinctive phenology, such as some fruit trees (e.g. durian). With December imagery the poor classification of sugar cane was not a surprise. At that time of year, the crop would either be standing very high ready for harvest and looking somewhat similar to surrounding bush grasses, or it would already have been cut and only the ratoons would be left. Textural measures to detect the regularity of fields under irrigation and additional knowledge such as factory locations, would probably improve the classification of such a crop substantially.

On the basis of these results it would not be sensible to recommend Landsat MSS imagery as a sole means of verifying current land use at the provincial level, since only a few crops could be accurately identified. Despite this the results are impressive for some crops, considering the simple processing applied and the low resolution data used. The average farm size in Rayong is 37 rai (6 ha., 1986 figures). Even without overlapping pixel boundaries this is covered by less than 10 pixels. Coupled with the fact that many farmers like to diversify their enterprise, with at least 2-3 different crops, this means that crop stands will on average be identified by the reflectance of only 2-3 pixels. Use of TM or SPOT data, at 30m or 10-20m resolution respectively, would significantly improve the accuracy for those crops identified and allow discrimination of others. This would certainly be true of TM data with it's improved spectral as well as spatial resolution. Because of high correlation between two bands on SPOT data the spectral resolution is actually quite poor, certainly in comparison to TM, but the high spatial resolution should be particularly useful if textural measures for crop identification were to be attempted. Spatial resolution of the data was the major constraint. The Thailand receiving station can now accept both SPOT and TM data, making such a means of provincial crop monitoring viable. In similar situations typical postal surveys of farmers, fail to predict crop areas more accurately than 30% either side of the estimate (priv. comm.). The current system of questioning only head farmers, is likely to be less accurate, so even the MSS results could be useful, particularly for such important crops to the Thai economy as paddy rice and cassava.

7.7.1.2 Incorporation of Ancillary Data

Video-grabbing proved to be a satisfactory method for digitisation of ancillary spatial information, although it was found to be as time consuming as use of a normal digitising tablet. Following a line on the screen is very similar to the way a line is followed with a digitising tablet, and the grabbed image must be of good quality to allow accurate discernment of boundaries. To achieve this the area to be covered usually had to be divided into at least four quadrants, to be video-grabbed separately in close-up focus. Each quadrant was then separately georeferenced and mosaiced with the rest. The quality of the video-digitised soil map image, was not good enough to allow automatic feature extraction.

The value of obtaining a high quality image was illustrated by the example of automatic feature extraction, achieved with the drum-scanned soil map image. This approach saves all the time otherwise required for line following. Once boundary lines and numbers were automatically obtained the process of colour coding, simply involved pointing with the mouse anywhere within a given boundary and typing in an appropriate colour value. The other advantage with the scanned image was that no mosaicing of images was required. Not all map information will be in such a convenient form for automatic feature extraction, but the representation of attribute boundaries in a highlighted colour, over a feinter topographic background is not uncommon. Unfortunately at present scanning costs tend to be prohibitively expensive, although costs vary substantially between firms and depending on the order pressure. Another means of avoiding the background detail on maps, is to locate the original print overlays used. Where this is possible a video-grabbed image would probably suffice. This can be taken a stage further where maps have been prepared by a digital process. For example the Soil Survey and Land Resource Centre of the UK, now have the soil maps they produce printed from digital data in raster form. As a bi-product of the printing process, the boundary information is available in raster format and could be easily incorporated into a GIS, or knowledge based image processing system.

7.7.1.3 The Addition of Soil Knowledge

The integration of soil information with the satellite image classification of Rayong had a substantial effect on area estimates, even though only paddy soil information was added. The results showed a doubling in the estimated area of paddy rice for amphoe Ban Khai. On comparison with the DLD estimate and that of the OAE, the new estimate appears to be closer to the true figure. Unfortunately no reliable error estimates for any of the figures were available, but even if the OAE estimate was as much as 50% out, the original classification seriously underestimates the true figure. It can be concluded that the addition of information on paddy soil has improved the spectral classification, but it is difficult to be sure of the true area of paddy rice for the 1981 season. The DLD estimate is likely to be closer than that of the OAE, as greater effort was expended on ground survey and air photo interpretation was used. This was because the DLD were preparing a map to last for several years, whereas the OAE were just providing their usual annual estimate. The evidence suggest that the OAE figure was an overestimate by at least 30%. Since the THAI government heavily subsidizes rice production in most years, accurate estimates of paddy land area are very important. Further investigation would clearly be needed before deciding to depend on estimates from satellite imagery, but the evidence suggest land utilisation estimates could be significantly improved, by reference to classifications of satellite image data.

Soil information is likely to influence the classification of other crops as well. For example, division between soil types showing excessive drainage and those exhibiting better water retention, was noted as particularly important in the distribution of fruit trees.

7.7.2 Mozambique Case Study

7.7.2.1 <u>The Significance of the Classification</u>

The classification produced provides a stratification of most of the province into different land cover areas of varying importance. The immediate result of most significance is the larger expanse of undeveloped land under forest, or dense savanna cover, particularly when contrasted with the Zimbabwe side of the border. The total area under class 1 and 3 (forest and dense savanna) of the land cover classification is approximately 17,000 Km^2 or about 30% of the province. Some of this will obviously be on land to steep to develop although, as can be seen from figure 7.3 below, in isolated pockets land is already under development in steep areas. This highlights another aspect, that of erosion risk. The classification has identified bare ground. In areas largely under forest cover, and on steep ground, where bare ground is indicated on the full resolution classification, there is likely to be cultivation of the type shown in plate 7.1.

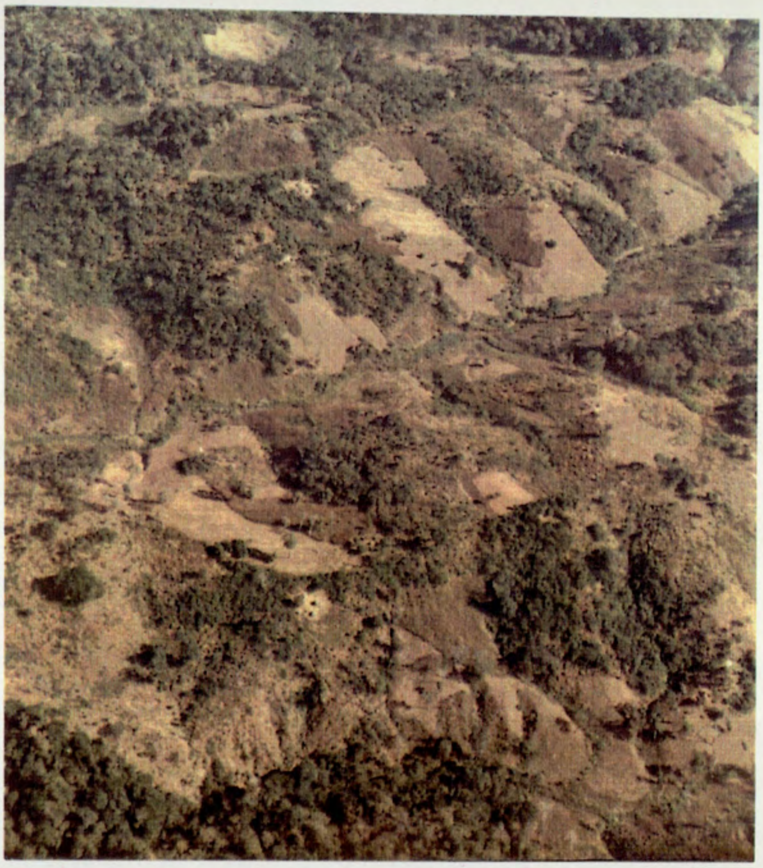


Plate 7.1 Illustrating how land is already under cultivation on quite steep slopes.

Confusion can arise from bare rock outcrops but since there is often a characteristic pattern to the cropping, it may be possible to detect the high risk areas, where extension advice will be needed. The cultivated area even with the bare rock areas included is very small, as can be seen on the provincial classification (see back pocket).

The general impression is of very little of the magenta class on the Mozambique side of the border in comparison to the obvious intensive utilisation on the Zimbabwe side. plate 7.2 showing a subsampled full scene also illustrates the sharp contrast well.

The classification could be improved if further ground survey information became available. Satellite image classification is really an iterative process, by which the classification is continually improved, through comparison of classification results with mapped ground data, refinement and then further comparison.

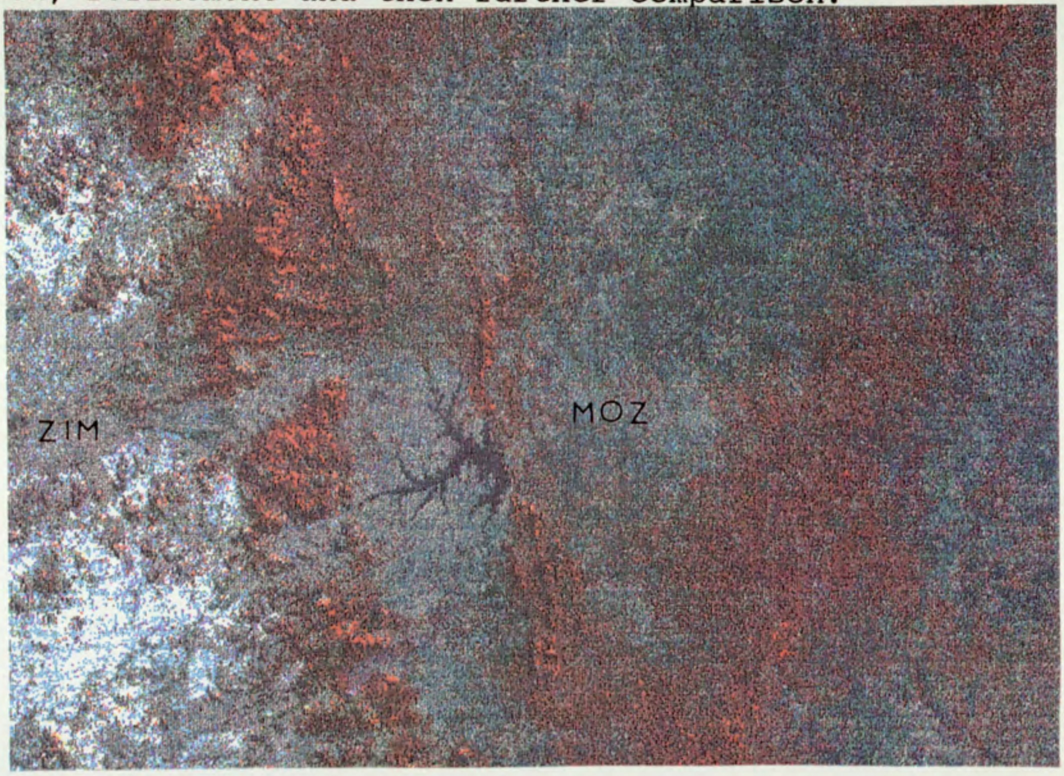


Plate 7.2 Subsampled full landsat scene at centre of Manica Province, illustrating the difference between the Zimbabwe and Mozambique sides.

Another option would have been to acquire high resolution satellite imagery for an area, probably the central scene again. This would allow the smaller parcels of land to be delineated and higher classification accuracy would be obtained. Also linear features such as main roads and railways would be more clearly visible, allowing vectorisation of these features from the imagery, so that roads and railways could for instance be superimposed on the image again, after a land cover classification. This would be an obvious advantage, if the intention were to produce results of a classification in a complete map form, with villages, roads, railways and other features, all shown together with land use. The Landsat MSS data used so far would not permit such feature extraction.

7.7.2.2 Commercial Adoption of Remote Sensing

The Mozambique satellite remote sensing study raised some important issues, concerning expectations clients have for remote sensing technology. The study also exemplified the misunderstandings current among other subject specialists. The perception of remote sensing technology, will obviously have a serious effect on the uptake of such methods, whatever the level of promotion from some agencies. In many ways the effectiveness of satellite remote sensing is still to be proved in the commercial sector.

In the Mozambique study three different and incompatible objectives, were current over the period of work, these were:

i) As understood by the remote sensing specialist -
To provide a land cover classification of a large, inaccessible province in Mozambique, at the reconnaissance scale, using Landsat MSS imagery.

ii) As understood by the client -
To use remote sensing as a solution to the problems of land use planning in an area where traditional sources of information were impractical. With the expectation that, solely from satellite imagery, detailed information on urban areas, infrastructure, slope and cropping at a detailed scale, would be available at high

accuracy.

iii) As understood by other project consultants - That from satellite imagery, a land use plan could automatically be derived without ancillary information and that processing time would be insignificant.

The intention of presenting the three different objectives, which were held throughout most of the remote sensing input, is not to suggest that any one in particular is correct. In the perception of each party, their objective was the correct one. This example shows that the introduction of satellite remote sensing and image processing, is not simply a question of refining and adopting technology, the technology and it's capability must be explained.

The aim of the Mozambique project, to produce an integrated rural development plan for Manica province, within a period of 5 months, is representative of many of the short term studies overseas. The period allotted for the remote sensing input was 32 days. On such short projects, channels of communication are often very indirect. This together with a requirement for the image processing to done in the U.K., allowed misconceptions to be perpetuated. In terms of the alternative options available, the final results of the satellite remote sensing analysis, provided useful information on land cover and a firm basis on which to proceed with further work. However, the client had undoubtedly expected a final product, in the form of a high quality map indicating current and proposed land use. Equally fellow consultants had anticipated obtaining more information from the imagery and seemingly a map of print quality. Most of the clients objectives could have been relatively easily achieved, but the input would have had to be about three times as long and the cost would probably have been, in the order of ten times the amount (judging from comparisons with other projects of this nature).

Costs would escalate because of the need to provide an on-site image processing system. Neither the client nor cooperating consultants on the Mozambique study, were happy to see so much time required for processing image data in the UK. The processing time would have been more acceptable, if the work was achieved at the project location, with interaction between the expert and local staff. The necessity of ancillary data would be more simply explained and it's effectiveness could be quickly proved. One of the main advantages of an on-site system would be that early results could be shown to the client for his comment and the technology and it's capabilities could be explained. Any misconceptions would be quickly eliminated and other consultants could also be involved, such that the possibilities for integration of different sector studies, by a computerised information system, would be self evident rather than an abstract concept.

Normally the only way to obtain a low cost DIPS on a contract is to cost the system into the job. This is why the cost of the Mozambique work would have escalated. For very short projects it may be difficult to justify such a cost, as the image processing component of a budget, then appears to dwarf some of the other inputs. For the Mozambique study, the decision to include such a system would be marginal. Yet a sophisticated information processing system offers advantages beyond the processing of satellite imagery. It offers the chance for integration of sector studies on a spatial and point data basis. This gives a framework within which results can be presented to the client, at all stages throughout the project. It also encourages a team of consultants to coordinate their activities and share the same information base, which each consultant helps to build. When presented in this way the provision of a low cost DIPS and associated software, can appear to be a much more cost effective option. For this general purpose role the DIPS should be installed early in the establishment of the project, as it will not be fully utilised if installed only part way through. Subject specialists must be involved with the development of an information system to serve the project, from the beginning of their input, otherwise their cooperation is unlikely.

The provision of the technology on-site is one means of improving the understanding of the client and other consultants, but it will often also be important to re-address the original proposal and talk directly to the client over what is expected. In the analysis of satellite remote sensing it is often sensible to examine only a small area in detail first and assess the advantages of further processing and purchase of imagery, on the basis of that example. This then gives an insight for the client into problems such as accuracy estimation. Since field data in detail is usually only available for very small areas anyway, a localised example study may be the only means of achieving any reliable accuracy assessment. The focus of interest is also likely to be confined to special areas of interest. In Manica province for example, the only 'safe' areas were about 5 Km either side of the Beira road, plus a very few of the larger villages outside the corridor. When flying over the province, landing anywhere outside Chimoio was not considered safe as the airstrip may have been under rebel control. In this situation although the whole province was of interest, the only areas where ground truth could be collected were in the Beira corridor. So in retrospect, a preliminary study of this area in detail might have been a more cost effective first approach. There was no opportunity to explain this to the client. Extracts from within archived SPOT and TM scenes are now being marketed and this may offer one possible means of a short low-cost, preliminary study, in the use of satellite remote sensing for a particular task.

7.8 Knowledge Capture and Representation

The Micro Expert study showed, that the use of a hierarchical structure, which has to be carefully planned and a reliance on one representation of data i.e. in terms of probability, was not appropriate to the construction of a complex land use planning model. Development was slow and in terms of the man-machine interface, the result was mediocre. The 'advice language' was not easily learnt, so that implementation of a system by a subject expert, would not be a trivial task. Such a system could not be recommended for any part of a land use planning system.

The interviewing of farmers showed that for Thailand, the best means of obtaining information on farmers social influences and eliciting local knowledge, was by means of an informal ad hoc interview with an expert. Such an interview, would then simply concentrate on capturing any relevant knowledge the farmer proffered, on selective questioning, as opposed to following a pre-defined pattern of questions. Interview time should not be fixed, so that if cooperation was poor, the interview would be brought to a swift conclusion, or if the farmer was helpful and knowledgeable, the interview would be prolonged.

ExpertEase illustrated that, with relatively few examples, induction systems can provide a decision rule that does appear to detect, some of the important limits or decision points. The importance of a comprehensive example set, with careful consideration of omissions, was identified. Categorisation of attribute values, partly to reduce the number of examples required, was found to be essential to avoid spurious results. Development of a system was very fast and the shell was easy to use. The interface was adequate, but relied upon a multiple choice presentation of possible attribute values. Selection was by means of keying the appropriate number, there was no natural language interface. It was not possible to add contextual help, which has been identified as an essential requirement of an expert system. It would be possible to decipher why a question was asked, by inspection of the rule screen, but the rule itself is difficult to understand because it is not displayed as a tree. The ExpertEase study, also indicated that for most of the previous ULG land use planning projects, insufficient example data was available for inductive systems to have been used successfully. It was not possible to communicate with other programs from ExpertEase, so despite some advantages, it could not be recommended as part of an automated land use planning system.

The application of SuperExpert, to land use planning problems in a typical overseas project environment, allowed comparison of the results with the known situation and other techniques for modelling the domain. Again, as with ExpertEase, categorisation of attribute values was found to essential to avoid spurious rules. Yet even with categorisation of the many different soil types in the Rayong study into five groups, all the available information was not utilised by the rule. Categorisation led to more factors being used in arriving at a decision, but decisions were still made on insufficient grounds, because the example set was too small. Because many possible situations were omitted, the algorithm assumed crop choice could in many cases be made, by testing just one or two attributes. Personal experience had shown that some of the attributes unused by the induced rule, were very important to farmer decisions.

The Yasothon study highlighted the problems involved in getting consistent information from a general questionnaire and how difficult it is to obtain the kind of information required for induction. This was because it would be difficult to justify a widespread survey, solely for specialist information, only of relevance to the induction exercise. On a typical consultancy job, there would not be time to collect the quantity of data, covered in the Yasothon survey. Lack of consistency in completion of forms, appeared to have been partly a result of using junior staff, who were unaware of the survey purpose, and may not have understood the importance, or nature of some questions. This would have to be considered in planning such a survey again, but usually there is little option to change this. Close inspection of the example set was required, to ensure representative examples were included. This indicates a reliance on expert judgement, despite the intention to arrive at decision rules without expert knowledge. Many decision points in the Yasothon data were unrealistic, again reflecting the need for an iterative process, whereby categories have to be adjusted to fit the available information, this is of course an expert task requiring judgement. With 289 examples, the Yasothon system was still an inadequate representation of the crop decision problem. It was possible to import information into the SuperExpert system, so the option exists for incorporation of such a shell, as part of a land use planning system, although large problems were found to require substantial disk space. The Yasothon problem was still a fairly simple problem by land use planning standards. The shell would not cope with larger problems with more attributes and examples. Considering the size constraint and the deficiencies in the decision rules, SuperExpert could not be recommended as part of an automated land use planning system.

This work showed that collection of the type of data, relevant to the derivation of a comprehensive land use plan, is not a trivial task. It cannot usually be accomplished by means of a single socio-economic survey, combined with physical data. The socio-economic setting can only be understood through detailed investigation by experts. A large number of factors are involved and many of these are very complex and often related. In other words data acquisition is the major problem and the inductive expert systems tested, were of no use in this phase of the exercise. In the case of land use planning in developing countries, it seems unlikely that it can ever be assumed, sufficient information will be available, to make use of inductive systems to model land use decision making. The inductive expert system shells, could have a minor role as decision support tools, to aid in an initial sorting of information by an expert, whereby key attributes and knowledge gaps are to be identified. The ease of use of such systems in terms of the man-machine interface and communication with other software, will be crucial to their adoption by subject experts.

The TIMM system showed that a rule-based expert system, was a form of knowledge representation well suited to the problem of crop selection and land use planning in general. As with the inductive systems actual example instances were used to train the system, but this time they were treated as rules. The symbolic, analogical partial match, inference engine seemed to be an appropriate means of interpreting the knowledge base, in that multiple crop selections were handled reasonably. The system still suffered the same handicap of insufficient information and although many answers were sensible, this was achieved partially by offering a wide range of crops, including some crops with a certainty of zero! At which level of certainty to exclude crops recommended, could only be decided from experience, which suggests the certainty factors should be treated merely as a very rough guide. The TIMM system was not as fast to implement as the induction systems, because data could not be imported and the in-built editor was very poor. Provided these deficiencies were corrected, such a system might be considered as a decision support tool, for representing current practice, enabling an expert to judge the likely action of farmers. However, unlike the SuperExpert shell, no indication is obtained of which factors are particularly important, or where knowledge is lacking. The shell is too restrictive in the number of parameters permitted, to be useful in building a land use planning model, even if only as a module within a larger system. The nature of the typical sub-problems envisaged, such as the farmer's choice, is too complex for the TIMM system.

Programming in the dBASE command language was similar to programming in any other procedural language, such as BASIC or C, although it is closer in syntax to the former. The advantage that dBASE has over the other procedural languages, is that the command language can be used to manipulate the databases defined within the DBMS, using a substantial syntax of functions and commands, specially designed for that purpose. In C or FORTRAN operations such as a `JOIN' or `SELECTION' would have to be programmed. The databases themselves can easily be defined and records entered, without recourse to programming expertise. So for applications demanding storage and retrieval of text, or numerical data, a DBMS is an obvious choice. The building of a part of a land use planning problem in dBASE, does demand programming expertise, but once the procedure has been coded (for example the matching algorithm discussed in section 7.5.3), the details on factors and crop requirements can be added or changed, without reliance on programming experience. This is because as with the Prolog system the procedural input/output operations were separated from the knowledge base. Like Prolog dBASE can also be used to automate data collection, by means of programmed data entry screens, but this would only be feasible where the data to be collected was already precisely defined.

The weakness of dBASE lies in it's lack of graphical presentation and poor file handling capability (no direct file access is provided and import/output of files is to a restricted format). Consequently a complete land use planning system is not easily built using dBASE, or a similar DBMS alone. Interfacing with graphics packages or other programming languages such as C for their graphics and file handling capabilities is essential. Speed of development of systems in dBASE was much slower than for expert system shells or Prolog, so where relational database provision is not a priority, the other IKBS options may be a better choice for fast prototyping.

The major advantage the Prolog language was found to have over the other IKBS studied, was it's declarative nature, whereby very little procedure has to be explicitly defined in the program. This had several consequences. Because rules could be added without significant alteration to procedural routines, all attributes did not need to be defined in advance. It is also the declarative style which facilitates fast prototyping of problems. The avoidance of fixed control, together with the near natural language syntax (by manipulation of symbols), enabled a high quality man-machine interface to be achieved, giving the user the impression of interacting with an intelligent system.

In contrast with dBASE, the Prolog version used had excellent graphic presentation and all necessary predicates for direct file access. The file access predicates meant image files could be queried or compared from within Prolog. Turbo Prolog (version 1.0) was compared with C (version 5.0) for speed, on a comparison of two input files with an answer written to one output file, showing that C was eight times faster, but the latest Prolog version appears to achieve adequate speeds on image manipulation, For major image processing tasks a language such as C is usually essential, but the facility to query image files quickly and with random access is a significant advantage of Prolog.

The ease with which the knowledge base can be developed in Prolog, creates the opportunity for an expert to eliminate written questionnaires or notes in knowledge acquisition, allowing him to capture the information direct to disk. This presumes the availability of a portable computer, but this is not unreasonable, as many overseas consultants have already come to depend on their own portable computer.

The Prolog approach to knowledge acquistion also assumes programming competence in the language. This is a major drawback as the language is not yet widely known beyond research circles. Most subject experts in overseas development work are only just tackling spreadsheets and databases. Thus a considerable commitment would be required on the part of a specialist to learn the Prolog language. For a version like Turbo Prolog this is comparable to learning C or FORTRAN althouhg the style of programming is very different. Rule-based expert systems which are user friendly have an advantage in this respect, as they can be swiftly applied without a siginificant learning period.

From the preceding discussion the essential characteristics of an IKBS for land use planning can be identified:

1/ The system should be rule based.

2/ The underlying structure should be declarative.

3/ The only size limitation to the system should be that imposed by the computer RAM/disk storage.

4/ The facility should be provided for direct call by another programming language and vice versa.

7.9 <u>The Role of Image Processing in Automation of Land Evaluation</u>

From the investigation of satellite remote sensing with Landsat MSS in the Rayong study in Thailand, MSS was found to be reasonably accurate in predicting large cropping areas of pineapple, cassava and paddy rice, compared to other available methods. Although significant commission occurred in all cases, because the spectral definition was not adequate. Even with a simple spectral classification the estimates were found to be of similar accuracy to current field survey methods. The use of TM imagery would improve both spectral and ground resolution. This would inevitably reduce errors of commission and also improve the overall accuracy figure, other crops might also be accurately identified. Paddy rice and cassava are particularly important to the Thailand economy, so accurate area estimates of these crops alone should justify the use of TM imagery. The cost effectiveness of using satellite imagery for crop area estimates on a regular basis was not estimated, but the results suggest a provincial level study would be worthwhile.

Automatic feature extraction of boundary lines from thematic maps was found to be relatively simple, provided the image obtained was of a high quality and the colours on the map were contrasting. Although the time saved was considerable, this would not be justified by the current cost, for small projects. The addition of ancillary information on soils was found to significantly improve the spectral classification. The study showed that incorporation of such data can lead to a very substantial improvement in classification accuracy. It should be emphasised that incorporation of soil data by means that simply treat it statistically, as another data set may not yield the same result. In the Rayong study a known relationship was exploited by logical inference, rather than attempting to investigate what relationships might exist.

In the Mozambique case study, satellite imagery was found to provide a useful means of stratification of a province, into different development areas for provincial planning.

The problems of verifying results under a hostile ground situation were evident. The client wished to know the accuracy of the land cover map produced, but no reliable means of establishing accuracy was evident. The use of aerial fly-overs to verify classes and explain them by means of standard 50 mm photographs was successful. The utility of the image processing results would have undoubtedly been improved by addition of slope information, as the significance of bare ground identified would have become evident Extraction of linear features from the Landsat MSS, such as railways and roads was not possible. The client was interested in such information, and particularly the location of villages. The results suggest that although useful information can be gathered, for initial stratification of a large inaccessible area, clients are likely to have higher expectations beyond what it is possible to achieve with MSS.

7.10 Conclusions

1/ Where computerisation of the land use planning process is the objective, the use of computers at the knowledge elicitation stage has advantages over conventional techniques. Social influences can be particularly important in making land use decisions. It is normally essential to acquire local knowledge on such effects for each new situation. An expert with a portable computer can offer significant improvement in the degree of knowledge elicited and a saving on the survey time required, compared to the traditional paper based interview.

The software used in such an interview should be a rule-based expert system, or a declarative programming language like Prolog.

2/ In the application researched inductive expert systems did not prove suitable for problems associated with land use planning because sufficient example data was not available. Where adequate data does exist, inductive systems can prove a useful tool for an initial appraisal of data to establish knowledge gaps and dependencies, however induced rules cannot be relied upon without further investigation.

3/ Rule-based expert systems provided a suitable form of knowledge representation for typical land use planning problems. Unfortunately all the systems tested proved inadequate in their capacity to model large problems, typical of a project situation. This type of constraint is however rapidly disappearing as new systems are released. The refinements added to the systems for expression of uncertainty were inadequate and misleading. No reliance could be placed on the certainty factors indicated. Such expert system shells will only be viable for part of the analysis, if they can directly access the common database, or call and be called directly through the main programming language. Applications should be identified and implemented by experts themselves wherever possible, rather than by a knowledge engineer with no land resource background.

4/ Declarative systems proved to be the most effective means of programming for the solution of typical land evaluation problems. For this either dBASE or Prolog proved suitable.

Although dBASE has a procedural command language, a declarative system can be built on the basis of the underlying databases. The dBASE system has the advantage that much data is already available in this format. The Prolog language has the advantage that if interview data is to be incorporated, this could be done directly from Prolog with little alteration. Many versions of Prolog also have inbuilt graphics facilities and allows for direct calls to and from other languages such as C and Pascal.

5/ In many cases the use of satellite remote sensing in a land use project would usually be difficult to justify on cost grounds, particularly since the precision of results for land cover is usually poor. Normally significant fieldwork is still required. In most instances image processing has to be done on site, to ensure participation by the local staff and other consultants, which adds to the cost. However, remotely sensed data may be the only source of up to date land use information.

Integration of other spatial data improves the land use classification sufficiently to show a significant gain over traditional methods of survey. If an image processing system is considered as providing a spatial analysis system for the coordination and monitoring of a complete land use planning exercise, the cost should normally be reasonably easily justified and quantified.

6/ In approaching computer automation on a land use planning project this research indicates that it is essential to adopt a declarative programming style. This will ensure flexibility, allowing the system to grow as knowledge is acquired and will encourage the integration of local knowledge, rather than the imposition of externally developed solutions.

Knowledge elicitation in the context of land use planning, requires experienced interviewers and often considerable local knowledge, which can only be accumulated over some time. The kind of information to be `captured' is of a socio-economic nature and it is not easily acquired by means of a standardized questionnaire.

Declarative or rule-based systems allow the acquired knowledge to be easily maintained. The knowledge base of rules is separate from programming procedure and can easily be altered without affecting the program. This is a very important advantage over conventional programs. Alterations to the knowledge base would not require expertise in programming. Thus as a social situation changed the model could be readily updated. Of course perception of such changes is still likely to require a specialist's knowledge. The database of factors concerning key indicators such as costs and prices can also be easily maintained by non-experts. Procedural changes to the actual programming method for the system would require expert attention in the case of a Prolog program, but could be accommodated by a subject specialist with an expert system shell. The adoption of a rule-based or declarative approach, should hopefully encourage awareness of the need to maintain evaluation programs to ensure they continue to be relevant. To this end it is important that those concerned with the final answer have some responsibility for monitoring and maintaining the rules.

7.11 <u>The Recommended Approach</u>

The most elegant and comprehensive solution to automation of a particular land use planning project, would be through a combination of programming languages used to develop a customised system. The preferred language for the majority of the system would be Prolog or a declarative language with similar merits. Using Prolog as the coordinating language would allow direct calls to C or a similar language for support of image processing and other numerical procedures. Alternatively for some projects C may be the better choice for coordination, with Prolog used to provide database and natural language facilities. This choice of design would require significant programming expertise demanding the assembly of a team of several programmers and system analysts. For a large land use planning project, phased over say three to five years, such an approach is recommended as time would be available for the establishment of an efficient and reliable system. For individual companies such projects unfortunately only occur occasionally.

For the more common situation where a land use planning system must be provided within say a six to eighteen month period, as part of a project with wider aims, there is unlikely to be any time for software development. Under such circumstances the implementation of a relational database system probably offers the best solution. The dBASE system would be the first choice as this capitalises on the widespread experience of consultants with this system. The disadvantage of using dBASE or similar relational databases is that most other programs cannot be called directly (only through DOS) and there is no provision for direct file access to image files as can be achieved with Prolog. A relational database is a reliable tool for solution of most land evaluation problems, but does not integrate well with raster based analysis systems. However, this situation changes rapidly and as of 1989 several relational databases are available with full language interfaces, which will run on small project based computers. They are still too expensive.

The great advantage of rule-based expert system shells over the other techniques investigated is their ease of use. Program maintenance can be achieved by a novice computer user, in contrast to a Prolog system. Equally because of the declarative nature of Prolog, program maintenance would not usually be required as such. Maintenace of the knowledge base would be simple by either mehtod. Local people with no previous skill could easily be trained to use expert system shells. Fewer individuals of the right calibre are likely to be available to master

Prolog sufficiently for program maintenance.

Provided the shortcomings of shells already mentioned, i.e. interfacing and capacity are resolved, it must be concluded that a rule-based expert system shell is the best option for handling the automation of social factors affecting land use planning. On long term projects the application of Prolog would probably be worthwhile because of the extra versatility which can be provided.

REFERENCES

ACKRILL, J.L. 1981. *Aristotle the Philosopher* Oxford:Oxford University Press

ADDIS, T.R. 1985 *Desigining knowledge-Based systems* London: Kogan Page

AGI 1989 AGI news in *Mapping Awareness* **2**:6

ALLAN, J.A. 1987. Strategies for Intelligent Classification of crops: sampling Ephemeral Data From Permanent Parcels in *Advances in Digital Image Processing* Proc. of the 13th Annual conference of RSS, Nottingham 7th-11th September 1987.

ALTY, J. L. and M.J. Coombs 1984 *Expert Systems-Concepts and Examples.* Manchester: NCC

ANDERSON, J.R., Hardy, E.E., Roach, J.T. 1972 *A land use classification system for use with remote-sensor data* United States Geological Survey Circular 671, Washington,D.C.

Ashton-Tate 1984 *dBASE III for your 16 bit PC* USA:Ashton Tate

BATE, J.St.J. and D.B. Vadhia 1987 *Fourth Generation Languages* Oxford: BSP Professional Books

BELWARD, A. and J. Taylor The use of multitemporal Landsat at data for improving crop mapping accuracy in Demen, M.C.J., Sicco Smit G. and Verstappen M.Th., *Remote sensing for Resources Development and Environment Management,* Rotterdam: Balkema pp 381-386

BEEK, K.J. and J. BENNEMA 1974 *Land Evaluation for agricultural land use planning-an ecological method.* FAO Soils Bull. **22**, 54-70

BIBBY, J.S. and Mackney,D. 1969 *Land Use capability classification* Tech. Mongr. No.1, Soil Survey of England & Wales:Harpenden

BIE, S.W. & J. Schelling 1977. An integrated information system for soil survey in *Factual Data Banks In Agriculture* Wageningen, Netherlands : PUDOC pp 45-53

BRINKMAN,R. and A.J. Smyth 1973. *Land Evaluation for Rural purposes* Publn. No 17, Wageningen: ILRI

BRADBURY, P.A. and A. Macdonald 1986 Accuracy assessment of land cover classification *Proceedings of a Remote Sensing Workshop 17th April 1986 at the Department of Geography, University of Nottingham, England.*, 126-132

BRAMER, M.A. 1985 Expert systems: The vision and the reality in *Research and Development in Expert Systems* edited by M.A. Bramer Press Synd. Univ. Cambridge:Cambridge

BRYANT, N.A. & A.L. Zobrist 1976. Integration of socioeconomic data and remotely sensed imagery for land use applications *Proc. 2nd Annual PECORA Symp.* ASP & USGS, Oct. 25-29, 1976 pp 120-1

BROONER, W.G. 1981 An Overview of Remote Sensing Input to GIS *PECORA III SYMP.- Remote Sensing: An input to Geographic Information Systems in the 1980's* Sioux Falls, Oct, 18-21 1981, edited by Richardson B.F. pp 318-329

BROONER, W.G. and Nichols, D.A. 1972. Considerations and Techniques for Incorporating Remotely Sensed Imagery into the Land Resource Managment Process in *Remote Sensing of Earth Resources* **1**, pp. 1-24, Tennessee:University of Tennessee

BUOL,S.W. 1972 Fertility Capability Classification system. *Agronomic-Economic Research on Tropical Soils, Annual Report for 1971*
Soil Sci.Dep., N.C. State Univ.:Raleigh, N.C.

CAMPBELL W.J. & M.Goldberg 1981. A 'user friendly' geographic information system in a color interactive digital image processing system environment. *Proc. VII ann. PECORA Symp.* ASP & USGS Sioux Falls, South Dakota, Oct. 18-21, 1981 pp 402-413

CODD, E.F. 1970 A Relational Model for Large Shared Data Banks. *Comm. ACM* **13** : 6,

COLMERAUER A. 1973 *Les systems-Que un Formalisme pour Analyser est synttesizer des Phrase sur ordinateur, Publiscation Interine No. 43,* Dept. d'informatique, Universite de Montreal, Canada.

COULSON, J.(edit) 1969 *The Little Oxford Dictionary*
Oxford: Claredon Press

COX, P.R. 1984. How we built Micro Expert in *Expert System Principles and case studies* edit. by R.Forsyth
London: Chapman and Hall

d'AGAPEYEFF, A. 1983 *Expert Systems*, Manchester: NCC

DAVIDSON, B.R. 1965. *The Northern Myth* Melbourne: Melbourne University.

DE GEUS, J.G. 1973. *Fertilizer Guide for the Tropics and Subtropics* Zurich : Centre d'Etude de l'Azote, pp774

DUDA, R.O., Gasching, J., Hart, P.E., konolige, K. Rebboh R, Barrett, P. & slocum, J. 1978, *Development of the PROSPECTOR consultation System for Mineral Exploration*, International Inc, Menlo Parke:C.A.

ERNST, G. and A. Newell, 1969. *GPS: a case study in Generality and problem Solving*, New York: Academic Press

ESA 1986. The spectral Signatures of objects in Remote Sensing *Proc of the Les Arcs, 1985 Symp. ISPRS Commision VII Working Group 3*, Paris: INRA/CNES.

ESRI 1984 *Map of Desertification Hazards - Explanatory Note* Redlands, California:ESRI

ESTES, J.E. 1984 Improved Information Systems: A Critical Need, *Proc of 10th Int. Symp on Machine Processing of Remotely Sensed Data* Indiana: Purdue Unversity West Lafaytte. pp 1-8.

FAO 1976. *A Framework For Land Evaluation Soils Bulletin No. 32* Rome:FAO

FAO 1983 *Guidelines: Land Evaluation for Rainfed Agriculture Bulletin No.52* Rome:FAO

FEIGENBAUM, E.A. and McCorduck, P.1984. *The Fifth Generation* New York : Michael Joseph

FISHER, H.T. 1979. *Thematic cartography Thematic Map Design* Harvard University Camb.,Mass. : Harvard University 8-38

FOLEY, J.D. and Van Dam, A. 1982 *Fundamentals of Interactive Computer Graphics* London:Addison-Wesley

FORSYTH, R. 1984 *Expert Systems* London : Chapman and Hall Ltd.

FRANK,T.D. 1981. Comprehensive geo-data base control with an electronic coordinate digitizer. *Proc. 7th Int. Sym. on Machine Processing of Remotely Sensed Data*, 6-23rd June 1981, Purdue University pp497-504

GITTINGER,J.P. 1972 *Economic Analysis of Agricultural Projects* London:The John Hopkins University Press

HARMON, P.,Maus,R. and Morrissey, W.1988 *Expert Systems Tools and Applications* New York:Wiley

HART, A. 1986 *Knowledge Acquisition for Expert Systems* London: Kogan Page

HARTNUP, R. and M.G. Jarvis 1976 *Soils in Civil Engineering and Planning* Soil Survey tech. Monog. No.13 110-34.

HAMLYN, D.W. 1970 *The Theory of knowledge* London:Macmillan

HAYWARD, S.A. 1985 Is a decision tree an expert system? in Research and Development in *Expert Systems* edited by M.A. Bramer Press Synd., Univ. Cambridge: Cambridge

HODGES, W.F. 1978. *Logic* London:Penguin.

HUNT,E.B.,Marin,J.& Stone,P.J.1966. *Experiments in Induction* London:Academic Press

ILACO, B.V. 1981. *Agricultural Compendium* Amsterdam : Elsevier, pp 739

INRA/CNES, 1982 and 1984. The Spectral Signatures of objects in Remote Sensing. *Proc. of the Avignon, 1982 and Bordeaux, 1983 Symp. ISPRS commission VII, working group 3*, Paris: INRA/CNES

ITL 1984 *Expert Ease User Manual* Glasgow:Intelligent Terminals Ltd.

ITL 1986 *Super Expert User Manual* Glasgow: Intelligent Terminals Ltd.

JACKSON,R.D. 1984. Remote sensing of vegetation characteristics for farm management in *Proc.of the Society of Photo Optical Instrumentation Eng.* **475**, 81-96

JEANSOULIN,R., E.Cals, & J.C.Darcas 1983. Multitemporal segmentation and analysis in remote sensing. *Proc. of the Geneva Meeting of the International Society for Optical Eng.*, pp 13-20

KANNEGIETER, A. 1985 Remote Sensing Applications for Land Use Planning in *The Applications of the FAO Framework for Land Use Planning in Thailand* Bangkok, Narathiwat-Thailand, 6-26 March 1985. pp 50-97

KASSAM, A.M., H.T. van Velthuizen, G.M. Higgins and A. Christoforides 1982 *Assessment of Land Resources for Rainfed Crop Production in Mozambique* Rome : FAO

KIRKPATRICK, E.M. (edit) 1983 *Chambers 20th Century Dictionary* Edinburgh:Chambers

KOWALSKI, R. 1983 Logic for Expert Systems in *Proc. of 3rd BCS Expert Systems Conference*

MARBLE, D.F. and D.J. Peuquet 1983 Geographic Information Systems and Remote Sensing. In R.N. Colwell(Ed) *Manual of Remote Sensing* (2nd ed.), Vol. I, Falls Church, Virginia : Am. Soc. of Photogrammetry pp 923-958

McKINION & H.E.Lemmon 1985. Expert systems for agriculture *Computers and Electronics in Agriculture* **1**, 31-40

McRAE, S.G. and C.P. Burnham 1981 *Land Evaluaton* Oxford: Clarendon Press

MEINDL, J.D. 1987 Chips for Advanced Computing *Scientific American* **257**, 4, 54-63

MERRITT,E.S., J.M.Meneely, L.Heitkemper & D.Hlavka 1981. CROPCASTTM : A special purpose geographically referenced information system for crop inventory applications. *Proc. of the 7th Int. Symp. on Machine Processing of Remotely Sensed Data* 6-23rd June 1981, Purdue University, 478-490

MICHIE,D.& Bratko,I.1978. Advice table representations of chess end-game knowledge. *Proc.AISB/GI Conf.*

MICHALSKI, R.S. and R.L. Chilausky 1980. Knowledge Acquistion by Encoding Expert Rules Versus Computer Induction from Examples. in *Int. J. Man-Machine Studies* **12**, 63-87.

MILLER, L.D., K. Nualchawee, C. Tom 1978 Analysis of the Dynamics of Shifting Cultivation in the Tropical Forests of Northern Thailand Using Landsat Modeling and Classification of Landsat Imagery *NASA TM 79545, Goddard Space Flight Center*, Greenbelt, Maryland USA

MILLINGTON, A.C. 1986 *Reconnaissance Scale Soil erosion mapping using a simple geographic information system in the humid tropics in Land Evaluation for Land-use planning in sloping areas* edited by W. Siderius ILRI pub'n 40, Wageningen : ILRI

MINSKY, M. 1974 A Framework for Representing Knowledge *The Psychology of Computer Vision* (P.Winston, ed.) pp 211-277, McGraw-Hill, London and New York.

MOOR, A.W., B.G. Cooke and L.G. Lynch 1981 *Information systems for Soils and Related Data.* Pudoc, Wageningen, The netherlands.

MOSS, R.P. 1968. Land Use, Vegetation and Soil Factors in South West Nigeria, a New Approach. *Pacific viewpoint* 107-26.

MOSS, R.P. 1969. The Appraisal of Land Resources in Tropical Africa: a critique of some concepts.*Pacific Viewpoint* **10**, 18-27.

MYERS, W.L. 1981. GIS Information Layers Derived From Landsat classification

NAYLOR, C.1983. *Build Your Own Expert System* Wilmslow: Sigma.

PATERSON, A., T. Niblett and A. Shapiro 1982. *ACLS User Manual.* Glasgow: Intelligent Terminals Ltd.

PEDLEY, M.I. 1986 Combined Remotely Sensed and Map Data as an aid to Image Interpretation and Analysis *Int.J.Remote Sensing* **7**, 3, 305-308

PEUQUET,D.J. 1983. The application of artificial intelligence techniques to very large geographic data bases *Proc. of the 6th Int. Symp. on Automated Cartography*, Ottowa, 1983 in *AUTOCAR* **6** 419-420

POWERS, J.E. et al. 1979. Advances in the development of remote sensing technology for agricultural applications. Preprint for Applications Session Symp., *13th Int. Astronautical Federation Congress, Munich, Germany, 16-23rd Sept. 1979*. Earth Observations Division, NASA : Houston, Texas

QUINLAN,J.R. 1979 Discovering rules by induction from large collections of examples. In *Expert systems in the micro-electronic age* Edited by D.Michie, 168-201 Edinburgh:Edinburgh University Press

RICHARDS,P. 1985. *Indigenous Agricultural Revolution* London: Hutchinson

ROGOFF, M. 1982. Computer Display of soil Survey Interpretation Using a Geographic Information System. *Soil Survey and Land Eva.***2**(2), 37-41

RSI 1980. *ORACLE Introduction* Menlo Park, Calif:Relational software Inc.

RUDEFORTH, C.C. 1975. Storing and processing data for soil and land use capability surveys. *J. Soil Sci.* **26** No.2 pp 155-168

SANCHEZ, P.A., W. COUTO and S.W. BUOL 1982 The Fertility Capability Soil Classification system: Interpretation, Applicability and Modification *Geoderma* **27**, 283-309

SCHULTINK, G.R. and W. Lodwick. 1981. Application of remote sensing and geographic information system techniques to evaluate agricultural production potential in developing countries. in *Proc. of the 7th Int. Symp. on Machine Processing of Remotely Sensed Data 6-23rd June, 1981 Purdue University* pp 466-477

SHAMMAS, N. 1987. *The New BASIC's* Redwood city, CA:M&T Publishing

SHEEHAN, D. 1979. A discussion of the SYMAP program in *Mapping Software and Cartographic Data Bases* Harvard University. Cambridge, Mass. : Harvard University pp167-179

SHORTLIFFE 1976 *Computer-Based Medical Consultations:MYCIN*, New York:Elsevier

SIMONS, G.L. 1985 *Expert Systems and Micros* NCC:Manchester

SIMONS, G.L. 1984 *Introducing Artificial Intelligence* Manchester:NCC

SINTON, D. 1979. An introduction to I.M.G.R.I.D., an information manipulation system for grid cell structures. in *Mapping Software and Cartographic Data Bases Harvard University.* Cambridge, Mass. : Harvard University pp 180-194

SOIL SURVEY STAFF 1975 *Soil Taxonomy* USA:USDA

SOIL SURVEY OF ENGLAND AND WALES (SSEW) 1984 *Soils and their use in Midland and Western England* Harpenden:Lawes Agricultural Trust

STEFIK, M. and CONWAY L. 1982. Towards the Principled Engineering of knowledge.*The AI Magazine* **3**,3, pp.4-16.

STERLING, L. and E. Shapiro 1986 *The Art of Prolog* London: MIT

TIMMNS, S.M., P.J. Ellis, and R.L. Bennetts 1984 Classification of Agricultural Cover Using Landsat and Aircraft Scanner Data *NZ Agricultural Science* **15** (4) pp. 209-216

TOU, J.T. and J.M. Cheng 1983. Design of a knowledge-based expert system for applications in agriculture. *IEEE Proc.,Trends and App. Automatic Intelligent Behaviour, Gaithersburg MO USA 25-26 May 1983* pp181-189

TROPSOILS PROJECT 1986 *An Expert system to Determine Lime Requirements for Soil of the Humid Tropics* Honolulu:University of Hawai

ULLMAN, J.D. 1982 *Principles of Database Systems.* Maryland, USA: Computer Science Press Inc.

USBR 1953 *Bureau of Reclamation Manual. Irrigated Land Use-Land Classification.* **5**, Part 2 Washington : US Dept,Interior

VALENZAUELA C.R., A.M.J. Meijerink and J. Huising 1986 Aerospace Imagery and Geo-Information systems for Watershed Management in *Proc. of 20th Int. Symp. on Remote Sensing of Environment, Nairobi, Kenga, 4-10 December1986.* pp Wageningen : Pudoc.

WELBANK, M 1983 *A Review of knowledge Acquisition Techniques for Expert Systems* Ipswich:Martlesham Consultancy Services

WEBSTER, R. 1977 An automated information system for soil survey Factual Data Banks in Agriculture Wageningen, Netherlands : PUDOC pp 54-56

WHITE, A.P. 1985 Inference deficiencies in rule-based expert systems in Research and Development in *Expert Systems* edited by M.A. Bramer Press Synd. Univ. Cambridge:Cambridge

WOOD,S.R.& Dent,.F.J. 1983 *A Land Evaluation Computer System Methodology* Bogor,Indonesia: Min.of Ag.Ind./FAO

YOUNG, A. 1976 *Tropical Soil and Soil Survey* Cambridge: Cambridge University Press.

YOUNG, J.A.T. and D.R. GREEN 1987 Is There Really A Role For Remote Sensing In Geographic Information Systems. in *Advances in Digital Image Processing*, Proc. of the Ann. Conf. of the Remote Sensing Society, Nottingham 7th-11th Sept. 1987. pp309-317.

ZADEH, L.A. 1965. Fuzzy sets *Information and Control* **8**, 338-353

GLOSSARY

286 & 386 refer to the 80286 and 80386 microprocessors which are the central processing units (CPU) used in IBM-compatible and IBM microcomputers. The 80286 is a faster processor than the 8088 used in the original PC and is found in PC-AT compatible models and the 80386 is in turn faster than the 80286. The 80286 and 80386 also add extra programming features to the CPU such as virtual memory and multi-tasking.

ANALOGUE Used in the context of varying electrical currents or voltages representing other quantities (light intensity) proportionally.

ARC INFO Vector information processing system(tradename)

BIT Binary Digit: The smallest unit of information in a digital device, it can have two possible values, ON(1) or OFF(0).

BYTE Grouping of bits (normally 8) handled as a unit by the computer.

C High level programming language (i.e. no direct correspondence between language instructions and machine code), with a suitable compiler it can be run on any machine.

CCD CAMERA This is a Charged Coupled Device camera meaning that an array of light sensitive sensors convert light into an electrical field which causes charges to be passed from one element of the device to the next.

FRAME-GRABBER A device for the digitisation of map or photographic images using a video camera linked to a computer.

GEO-REFERENCE Information is referenced to some form of geographic coordinate system.

HARDWARE Physical units making up a computer system.

LAND UNIT A discrete area of land with uniform attributes (i.e. factors of importance to land use).

LOOK-UP TABLE A look-up table (LUT) is a hardware mechanism used to transform pixel values. An input pixel value is transformed to a new value.

MACRO A list of commands to be executed in sequence, which can form a complete program.

MOUSE A device which allows the position of a cursor on a VDU screen to be controlled by moving the 'mouse' across a flat surface.

NORMALISE Adjust to a common scale.

OPERAND The object upon which a particular operation is performed by an operator such as a relation like AND or NOT.

PIXEL A discrete picture element in a two dimensional array of such elements which forms a raster image. Each pixel in the image has a specific intensity or colour.

SOFTWARE Any type of program.

USER-FRIENDLY Appealing to a user with no previous computing expertise.

APPENDIX A

MICRO EXPERT

An example of the output from the micro expert sytem written is given below:

```
Wetness class, type A for details of how to answer  answered with 3.00


Depth to impermeable layer type A for details of how to answer  answered with 3.00


The volumetric retained water capacity as a percentage type A for details

of how to answer  aswered with 35.00


                     -5 -4 -3 -2 -1  0  1  2  3  4  5

001 poaching is likely                           A


Danger of poaching land is

Very Likely
```

The numeric range displayed between -5 and +5 is the histogram used to represent probability. A probability of -5 represents zero and +5 represents a probability of 1.

PROGRAM LISTING FOR MICRO EXPERT EXAMPLE

```
PREAMBLE 'This is a simple assessment of land trafficability'
GOAL POACHINGRISK 'Danger of poaching land is
BRIEF 'poaching is likely'
BAYESIAN VERYLOWTRAFF LS 2.00 LN 1.00
         LOWTRAFF     LS 1.80 LN 1.00
         MODTRAFF     LS 1.60 LN 1.00
         HIGHTRAFF    LS 1.40 LN 1.00
         VHIGHLYTRAFF LS 1.20 LN 1.00
PRIOR 0.5
RULE VHIGHLYTRAFF 'There is a low risk of poaching'
TRACE 'Considering climate, soil depth and drainage status'
AND WELLDRAINED DEEP DRYSOIL
RULE HIGHTRAFF 'The land has a high trafficability'
TRACE 'considering climate, soil depth and drainage status'
OR TRAFFICABLE1 TRAFFICABLE2
RULE MODTRAFF 'The trafficability of the land is moderate'
OR TRAFFICABLE3 TRAFFICABLE4
RULE LOWTRAFF 'The trafficability is low'
OR TRAFFICABLE5 TRAFFICABLE6
RULE VERYLOWTRAFF 'The trafficability is very low'
OR TRAFFICABLE7 TRAFFICABLE8
RULE WELLDRAINED 'The land is well drained'
TRACE 'Considering quantitative data on soil moisture'
MODULUS MULTICHOICE 1.0 0.01
RULE IMPERFECTPOOR 'The land is imperfectly or poorly drained'
TRACE 'Considering quantitative data on soil moisture'
MODULUS MULTICHOICE 2.0 0.01
RULE VERYPOOR 'The land is very poorly drained'
TRACE 'Considering quantitative data on soil moisture'
MODULUS MULTICHOICE 3.0 0.01
QUESTION MULTICHOICE 'Wetness class, type A for details of how to answer

AMPLIFY 'Type the number that corresponds to the wetness class: '
'1 for class I OR II; 2 for class III or IV; 3 for class V or VI'
NUMERIC 1.0 3.0
RULE DEEP 'Depth to impermeable layer of soil >80cm'
TRACE 'considering depth to indurated horizon from soil description'
MODULUS MULTICHOICE2 1 0.01
RULE MODEEP 'Depth to impermeable layer of soil between 40-80cm'
TRACE 'Considering depth to indurated horizon from soil description'
MODULUS MULTICHOICE2 2 0.01
RULE SHALLOW 'Depth to impermeable layer of soil less than 40cm'
TRACE 'Considering depth to indurated horizon from soil description'
MODULUS MULTICHOICE2 3 0.01
QUESTION MULTICHOICE2 'Depth to impermeable layer'
' type A for details of how to answer'
AMPLIFY 'Type the number that corresponds to the depth to '
'impermeable horizon: '
'1 for >80cm; 2 for 40-80cm; 3 for <40cm'
NUMERIC 1.0 3.0
RULE DRYSOIL 'High Available water capacity'
RANGE VRWC 35 45
QUESTION VRWC 'The volumetric retained water capacity as a percentage'
' type A for details of how to answer'
AMPLIFY '45% or above should be input as 45,simarly for'
'35% or below'
NUMERIC 35 45
RULE TRAFFICABLE1 'I/II,40-80cm,low moisture capacity'
AND WELLDRAINED MODEEP WETSOIL
RULE TRAFFICABLE2 'III/IV, >40cm,moderate or low moisture capacity'
AND IMPERFECTPOOR NOTSHALLOW WETSOIL
RULE NOTSHALLOW 'Depth to impermeable layer >40cm
```

```
62      NOT SHALLOW
63      RULE TRAFFICABLE3 'The drainage class is III/IV, and the soil is'
64      'mod. shallow with a low moisture capacity'
65      AND IMPERFECTPOOR NOTDEEP WETSOIL
66      RULE TRAFFICABLE4 'The drainage class is III/IV,and the soil is'
67      'mod. shallow with a high moisture capacity'
68      AND IMPERFECTPUOR NOTDEEP DRYSOIL
69      RULE TRAFFICABLE5 'The soil is of drainage class III/IV'
70      'very shallow & low moisture capacity'
71      AND IMPERFECTPOOR SHALLOW WETSOIL
72      RULE TRAFFICABLE6 'Very poorly drained deep soil,high moisture capacity'
73      AND VERYPOOR DEEP DRYSOIL
74      RULE TRAFFICABLE7 'Very poorly drained deep soil,low moisture capacity'
75      AND VERYPOOR DEEP WETSOIL
76      RULE TRAFFICABLE8 'Very poorly drained & moderately shallow'
77      AND VERYPOOR NOTDEEP
78      RULE NOTDEEP 'Depth to impermeable layer <80cm'
79      NOT DEEP
80      RULE WETSOIL 'Very low available water capacity'
81      NOT DRYSOIL
82


list of GOALS
  1     POACHINGRISK.......................   2


list of HYPOTHESES
  1     POACHINGRISK.......................   2
  2     VERYLOWTRAFF.......................  20
  3     LOWTRAFF...........................  18
  4     MODTRAFF...........................  16
  5     HIGHTRAFF..........................  13
  6     VHIGHLYTRAFF.......................  10
  7     WELLDRAINED........................  22
  8     DEEP...............................  35
  9     DRYSOIL............................  50
 10     TRAFFICABLE1.......................  57
 11     TRAFFICABLE2.......................  59
 12     TRAFFICABLE3.......................  63
 13     TRAFFICABLE4.......................  66
        TRAFFICABLE5.......................  69
 15     TRAFFICABLE6.......................  72
 16     TRAFFICABLE7.......................  74
 17     TRAFFICABLE8.......................  76
 18     MULTICHOICE........................  31
 19     IMPERFECTPOOR......................  25
 20     VERYPOOR...........................  28
 21     MULTICHOICE2.......................  44
 22     MODEEP.............................  38
 23     SHALLOW............................  41
 24     VRWC...............................  52
 25     WETSOIL............................  80
 26     NOTSHALLOW.........................  61
 27     NOTDEEP............................  78

This Model Does Not Use Externals

        Cross-Reference Listing

Line      No Name                                  Used By
  35       8  DEEP                                   VHIGHLYTRAFF NOTDEEP TRAFFICABLE6
                                                 TRAFFICABLE7
  50       9  DRYSOIL                                VHIGHLYTRAFF WETSOIL TRAFFICABLE4
                                                 TRAFFICABLE6
```

APPENDIX B

INDUCTIVE SHELLS

The Theoretic Measure for Attribute Splitting As Used in the ACLS induction algorithm of Super Expert.

The following account is taken from the "ACLS USER MANUAL" (Paterson et al., 1982).The choice of an attribute on which to split at any node is performed using an information theoretic measure, to be described below. The attribute which has the "best" value according to this measure is used.

The algorithm used by Super Expert works rather differently for Integer-valued attributes than for logical attributes. We consider the logical attributes first.

At any node we have a set of N attribute vectors. Let us consider a simple example of what can happen. Assume there are two class-values, C_1 and C_2 and assume that the attribute under consideration takes two values V_1 and V_2. If we split our N attribute we would obtain the following 2 x 2 matrix.

$$\begin{array}{c|cc} & C_1 & C_2 \\ \hline V_1 & P_{11} & P_{12} \\ V_2 & P_{21} & P_{22} \end{array}$$

In this case P_{ij} is the probability of one of the attribute vectors having a value V_i for the attribute and class value C_j. We can now define the entropy of the initial state before splitting on the attribute. This is given by

$$-((P_{11}+P_{21}))\ \ln\ (P_{11}+P_{21})+$$
$$(P_{12}+P_{22})\ \ln\ (P_{12}+P_{22}))$$

By splitting on the attribute, it is easy to see that we have decreased the entropy, which is now:

$$-(P_{11}\ln(P_{11}) + P_{12}\ln(P_{22})$$
$$+ P_{21}\ln(P_{21}) + P_{22}\ln(P_{22}))$$

The difference (initial Entropy - final Entropy) is a measure of the usefulness of the attribute. The attribute with the highest value of this difference is the one on which we split. In the case above this difference is

$$p_{11}\ln(p_{11}/(p_{11}+p_{21}))+$$
$$p_{12}\ln(p_{12}/(p_{12}+p_{22}))+$$
$$p_{21}\ln(p_{21}/(p_{11}+p_{21}))+$$
$$p_{22}\ln(p_{22}/(p_{12}+p_{22}))$$

This example is easily generalised. If we have M class values and N possible values for a particular attribute then the measure to be maximised is

$$\sum_{i=1}^{N}\sum_{j=1}^{M} P_{ij} \ln \left[\frac{P_{ij}}{\sum_{k=1}^{M} P_{ik}} \right]$$

Again P_{ij} represents the probability of i_{th} value of the attribute occurring with the j_{th} class value.

With an integer attribute we not only need to calculate the value of the measure but we also have to compute the value on which to split the attribute.

Let us assume that we have N attribute vectors $V_1,\ldots.,V_N$ and that we are concerned with attribute A. We can further assume that the attribute vectors are ordered in increasing values of A.

For each $i(1<i<N)$ we can split a vector into two subsets, $\{V_1, \ldots, V_i\}$ and $\{V_{i+1} \ldots, V_N\}$. These subsets define a value of the evaluation function defined above. We choose the minimum such value as the value of the evaluation function for A. If this value occurs for subsets $\{V_1, \ldots, V_j\}$ and $\{V_{j+1}, \ldots, V_N\}$ then we split the attribute at a value which is midway between the values at V_j and V_{j+1}.

For logical attributes, when several of these have the same entropy value, the attribute normally chosen, to split on, is the one which appears first in the attribute display order for the sub-problem.

Results from checking example set against the SuperExpert system

The 29 examples used for verification were not used in training the expert system. Clashes where more than one outcome occurs are explained in brackets [].

CORRECT	Quest. No.	CROP CHOICE	ACTUAL CHOICE
Y	C1	Rubber	Rubber(30 rai) Durian, Rambutan, cassava and groundnut.
N	C2	Mango	Durian and Rambutan
Y	C3	Clash [Rubber-cocount-Paddy]	Rubber, durian and paddy
N	C4	Mango	Sugar cane, Rubber and Durian
Y	C5	Cassava	Cassava

N	C6	Null	Rubber and Bananas
Y	C7	Cassava	Cassava and Paddy
N	C8	Null	Cassava and Coconut
Y	C9	Paddy rice	Coconut, Cassava and Paddy
Y	C10	Clash	Paddy, Coconut, Casssava and durian.
N	C11	Paddy Rice	Mangosteen, Durian and Cassava
N	C12	Paddy Rice	Jub Jub
N	C13	Paddy Rice	Lamud, Coconut and runner beans.
N	C14	Clash [Durian-Rambutan-Mangosteen-Coconut-Mango] [after one answer of durian to OTHEXP]	Durian, Coconut, Paddy and Cassava
N	C15	Rubber	Cassava
Y	C16	Clash [Coconut-Cassava-Rubber]	Cassava-pineapple-coconut.
Y	C17	Clash [Coconut-Cassava-Rubber]	Cassava and coconut
N	C18	Rubber	Cassava, pineapple, Jackfruit and Coconut.

N	C19	Rubber	Eucalyptus, Pinus, Sugar-cane, Cassava and pineapple.
N	C20	Mango,Rubber,	Pineapple and Durian.
N	C21	Mango	Pinapple-Durian-Rambutan
Y	C22	Paddy rice	Paddy rice
N	C23	Cassava	Cashew-Sathorn- Mango-Jackfruit
	(The slope was B/C but this was never queried, the land would be unsuitable for cassava)		
N	C24	Mango	Sugar cane-cassava
	(Mango would grow quite well but would not be a sensible choice as the sole crop because of low return and variable market.)		
Y	C25	Rubber	Rubber, Asparagus, Cassava
N	C26	Mango	Durian-Sapodilla Mangosteen-Rubber
N	C27	Null	Rubber
	(Null was given after answering C to slope query)		
N	C28	Coffee	Durian
	(With 7 rai land labour requirement was met from within family)		
N	C29	Coffee	Durian
	(Questions asked were on previous experience, capital and moisture regime.)		

LISTING OF RAYONG2 DATA

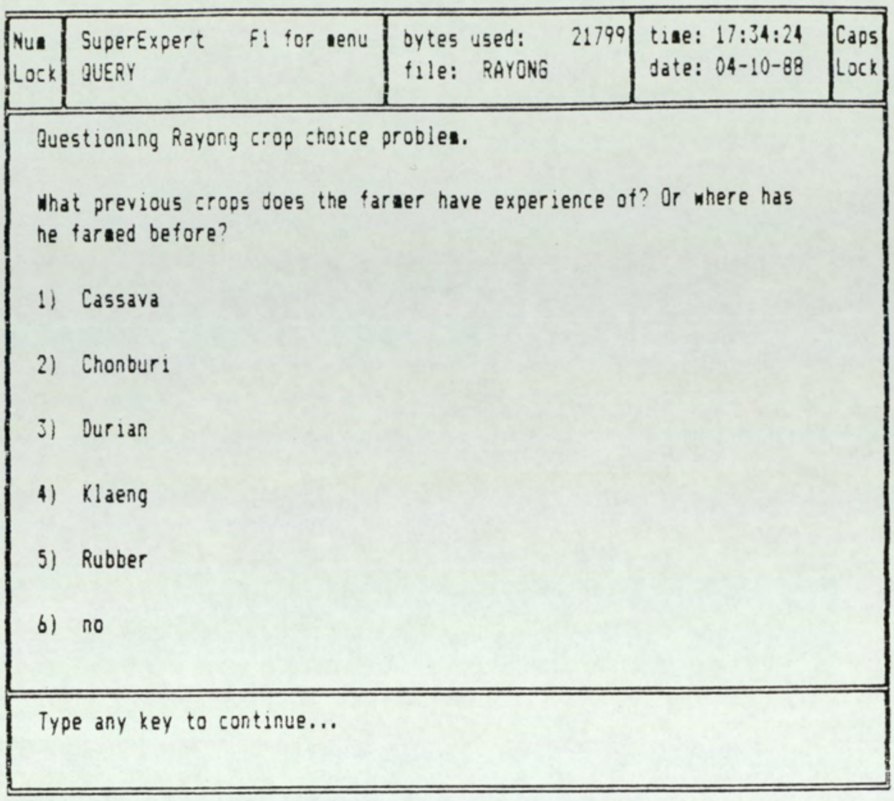
MOIST	SERIES	SLOPE	AGE	TEN	AREA	EXP	OTHEXP	EMP	CAP	FAMLAB	HIRATE	OXEN	TRACT	TILLER	WATER	PUMP	LOAN	PRIV	BANK	CROP
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	CASSAVA
F	PIEDMONT	B	2	P	50	4	Klaeng	F	B	F	50	T	T	F	8	F	T	F	T	CASSAVA
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	CASSAVA
F	UNDULATING	B	2	F	20	6	Chonburi	F	C	T	40	T	T	F	3	F	F	F	F	CASSAVA
F	EROSION	ABD	1	F	29	20	Paddy rice	F	B	T	30	T	F	F	6	T	F	F	F	CASSAVA
F	UNDULATING	AB	1	F	9	5	Cassava	F	B	T	40	T	T	F	3	F	F	F	F	CASSAVA
F	ALLUVIAL	A	2	F	2	24	no	F	C	T	40	F	F	F	2	F	F	F	F	CASSAVA
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	CASSAVA
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	COCONUT
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	COCONUT
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	COCONUT
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	COCONUT
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	COCONUT
F	UNDULATING	A	3	F	38	50	Paddy rice	F	B	T	40	T	F	T	2	T	F	F	F	COCONUT
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	COCONUT
T	PIEDMONT	B	2	F	48	20	no	T	A	F	46	F	F	F	2	T	F	F	F	COFFEE
T	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	COFFEE
T	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	COFFEE
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	DURIAN
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	DURIAN
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	DURIAN
F	EROSION	BDC	2	F	30	7	Rubber	T	A	T	40	F	T	F	5	T	F	F	F	DURIAN
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	DURIAN
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	DURIAN
F	PIEDMONT	B	2	F	48	20	no	T	A	F	46	F	F	F	2	T	F	F	F	MANGO
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	MANGO
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	MANGO
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	MANGOSTEEN
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	MANGOSTEEN
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	MANGOSTEEN
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	PADDY RICE
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	PADDY RICE
F	ALLUVIAL	A	1	F	14	35	no	F	B	F	40	F	F	T	3	F	T	F	T	PADDY RICE
F	ALLUVIAL	A	3	F	4	61	no	F	B	T	40	T	T	F	2	T	F	F	F	PADDY RICE
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	PADDY RICE
F	UNDULATING	A	3	F	38	50	Paddy rice	F	B	T	40	T	F	T	2	T	F	F	F	PADDY RICE
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	PADDY RICE
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	RAMBUTAN
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	RAMBUTAN
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	RAMBUTAN
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	RAMBUTAN
F	EROSION	B	2	F	14	20	no	F	B	F	77	T	T	F	6	F	T	F	T	RUBBER
F	EROSION	C	1	P	41	21	Rubber	F	A	T	44	F	F	F	9	F	F	F	F	RUBBER
F	PIEDMONT	B	2	P	50	4	Klaeng	F	B	F	50	T	T	F	8	F	T	F	T	RUBBER
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	RUBBER
F	PIEDMONT	B	2	F	50	15	Rubber	F	A	F	66	F	F	F	4	F	F	F	F	RUBBER
F	EROSION	B	2	F	100	20	Cassava	F	A	F	48	F	F	F	7	F	T	F	T	RUBBER
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	RUBBER
F	UNDULATING	B	2	F	21	32	Paddy rice	F	B	T	35	T	T	F	4	T	T	T	F	RUBBER
F	ALLUVIAL	A	1	F	14	35	no	F	B	F	40	F	F	T	3	F	T	F	T	RUBBER
F	EROSION	ABD	1	F	29	20	Paddy rice	F	B	T	30	T	F	F	6	T	F	F	F	RUBBER
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	RUBBER
F	UNDULATING	A	3	F	38	50	Paddy rice	F	B	T	40	T	F	T	2	T	F	F	F	RUBBER
F	EROSION	BDC	2	F	30	7	Rubber	T	A	T	40	F	T	F	5	T	F	F	F	RUBBER
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	RUBBER

KEY TO RAYONG2 LISTING

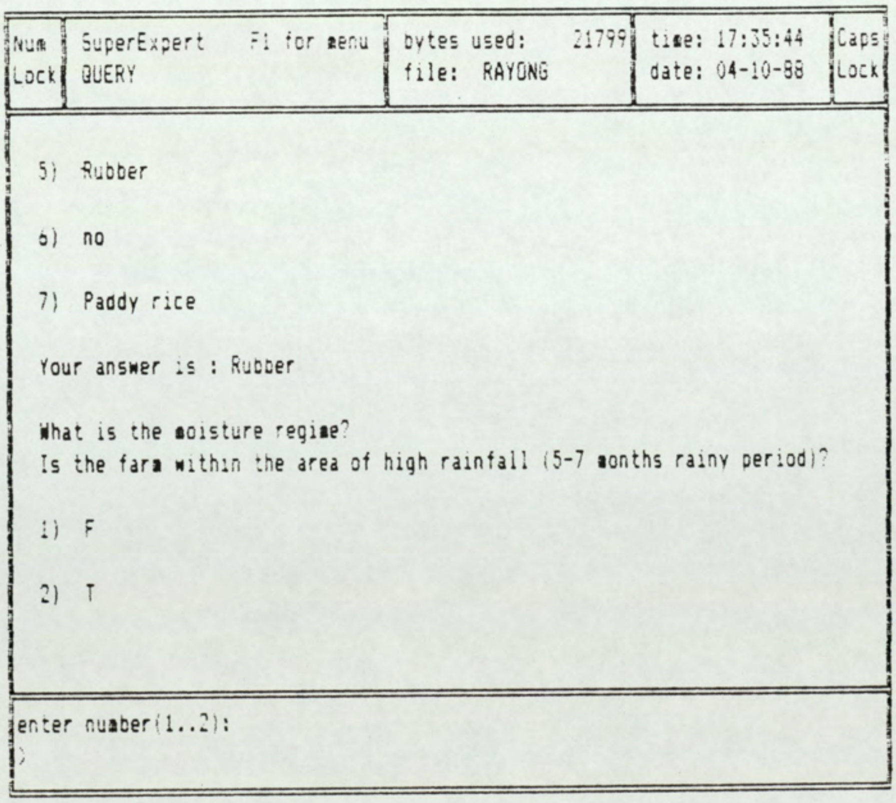
MOIST(True or False)	Moist agroclimatic regime or not
SERIES	Soil Series grouping
AGE	Age group of farmer 1,2, or 3.
TEN	Tenure status - (F)reehold, (P)ermanent, or (T)emporary?
AREA	Area of the farm in rai(1600m^2)
EXP	Years farming
OTHEXP	Other locations or crops farmed previously.
EMP	Other employment ie part-time.
CAP	Classification of wealth
FARMLAB	Family labour used.
HIRATE	Local cost (daily rate) of hiring labour.
OXEN	Oxen used for cultivation True or False.
TRACT	Tractor used for cultivation.
TILLER	Power tiller used for cultivation
WATER	Depth in metres to water table at beginning of dry season.
PUMP	Water pump available.

LOAN	In debt, true or false.
PRIV,BANK	Loan from private source or a bank.
CROP	Crop grown.

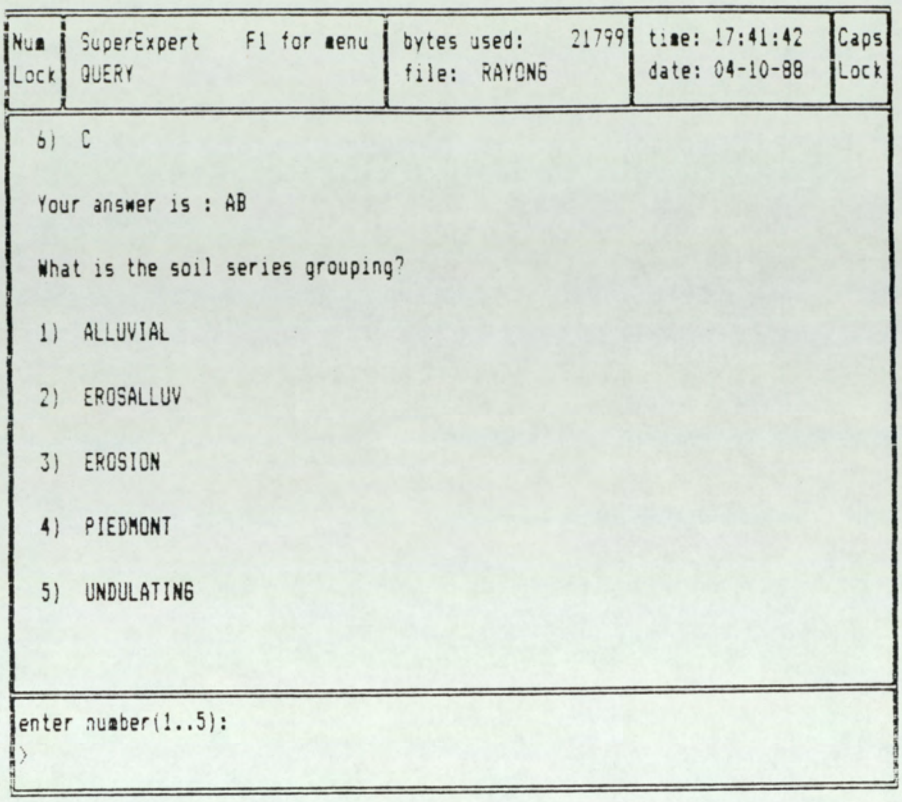
The following two pages show an example of an interactive query with the system.



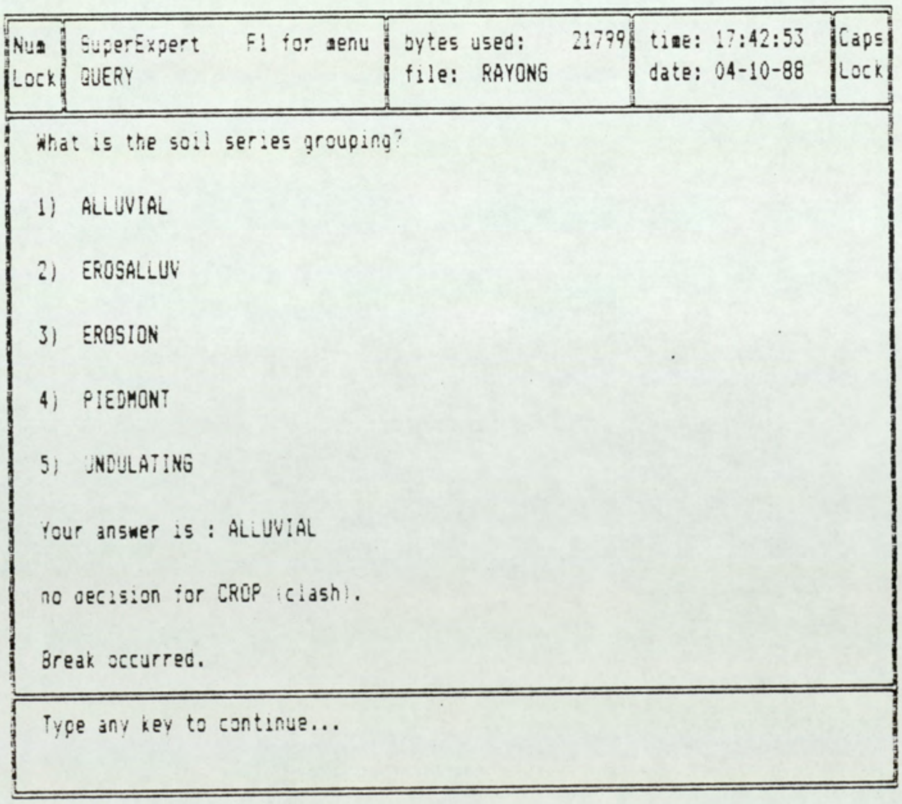
```
Num   SuperExpert    F1 for menu   bytes used:     21799   time: 17:34:24   Caps
Lock  QUERY                        file:  RAYONG           date: 04-10-88   Lock

 Questioning Rayong crop choice problem.

 What previous crops does the farmer have experience of? Or where has
 he farmed before?

 1)  Cassava

 2)  Chonburi

 3)  Durian

 4)  Klaeng

 5)  Rubber

 6)  no

 Type any key to continue...
```



```
Num   SuperExpert    F1 for menu   bytes used:     21799   time: 17:35:44   Caps
Lock  QUERY                        file:  RAYONG           date: 04-10-88   Lock

  5)  Rubber

  6)  no

  7)  Paddy rice

  Your answer is : Rubber

  What is the moisture regime?
  Is the farm within the area of high rainfall (5-7 months rainy period)?

  1)  F

  2)  T

enter number(1..2):
>
```



```
Num  SuperExpert    F1 for menu   bytes used:     21799   time: 17:41:42   Caps
Lock QUERY                        file:  RAYONG           date: 04-10-88   Lock

  6)  C

  Your answer is : AB

  What is the soil series grouping?

  1)  ALLUVIAL

  2)  EROSALLUV

  3)  EROSION

  4)  PIEDMONT

  5)  UNDULATING

enter number(1..5):
>
```



```
Num  SuperExpert    F1 for menu   bytes used:     21799   time: 17:42:53   Caps
Lock QUERY                        file:  RAYONG           date: 04-10-88   Lock

  What is the soil series grouping?

  1)  ALLUVIAL

  2)  EROSALLUV

  3)  EROSION

  4)  PIEDMONT

  5)  UNDULATING

  Your answer is : ALLUVIAL

  no decision for CROP (clash).

  Break occurred.

  Type any key to continue...
```

Description of rules induced in Super 3 expert system (From examination of the Rule listing screen) and notes on the relevance of induction to the problem of knowledge elicitation from farmers (made on initial inspection of results).

PRIORITY ERROR

1. If 'Previous Experience' in klaeng Then grow Cassava OR Rubber

Comment: Because only one farmer questionned had experience in the klaeng district and he was growing Rubber and Cassava this attribute value has been given undue weight, emphasising the importance of trying to ensure all possible outcomes associated with a value are included in the data set. In practice this requires a very close monitoring of question replies which is not feasible in a field situation.

INADEQUATE DATA SET

2. If 'Previous Experience' is in Cassava
   AND 'Soil Series' is ALLUVIAL
   THEN NO ANSWER

Comment: After 'previous experience', 'soil series' group has been identified as the attribute of the next highest priority. However, in this case no example was available for the combination of previous experience growing cassava and the ALLUVIAL soil series group, therefore no answer could be provided. This is a perfectly legitimate combination, but certainly requires further information from other attributes. Such information was available in the example set for combinations associated with the ALLUVIAL grouping but not with a previous experience in cassava also associated, thus the factor of previous experience OTHEXP is dominating the analysis with a higher priority accorded to it than any other factor. This suggests an inadequacy in the algorithm applied.

SIMPLISTIC ASSUMPTION

3. If 'Previous Experience' is in Cassava
   AND 'soil series' is UNDULATING
   THEN grow CASSAVA

Comment: A reasonable chuice which may not be wrong but the conclusion is based on inadequate information, most if not all of the other factors should have been considered. For instance if the land is steeply sloping (say greater than class B) CASSAVA should not be grown, besides which if other conditions prevail a more profitable crop such as DURIAN might be grown. Because the two attributes of OTHEXP AND SERIES can be applied with the given values to match with a unique outcome of CASSAVA in the data set no other attributes are considered. With a larger sample set there is less chance of a unique occurrence of a crop with a limited combination of attribute values. Unfortunately with something over 1.5 million combinations to be covered in this particular problem there is little chance of collecting a reasonable data set in reasonable time or within a sensible budget.

The kind of rule formed above is particular dangerous in development of an expert system as a 'knowledge engineer' without subject expertise may not detect the anomaly and could in principle easily accept a totally fallacious rule set. Presuming a classic procedure of validation was followed by reference back to a subject expert then such anomalies should be detected and one would expect such a simplistic rule as that above to be rejected. There remains the doubt that in more borderline cases with more attributes used, but nevertheless an inadequate specification, an expert on being asked if the crop suited the characteristics may say yes, on the basis that the crop was not unsuited. Such doubts become important when considering the value of experts time and the reluctance as experienced in Thailand, to express too definite an opinion. All too often it will be left to the knowledge engineer to use his judgement in such a position the expert systems of this nature cannot be said to be a very reliable support to the decision, on the contrary the rule tends to hide the facts.

Rather than working through the complete rule list for which most comments would be on a similar basis to those given for the preceding 3 rules, the rule base established for the prolog system was checked to see if any matches can be found in the SUPER 3 listing.

From FARMREC. PRO

1st rule:

```
PADDY RICE  :-  PADDY LAND

                AND NO WATER CONTROL
```

This is matched by

ALLUVIAL > PADDY RICE

Note, WATER <3, CAPITAL=B and NO PREVIOUS EXPERIENCE were all factors considered first, but no account has been taken of the presence of water control. Yet factors such as depth to water table and possession of a pump were included. This is another example of SIMPLISTIC ASSUMPTION. The error may have occurred for two reasons. One reason is that without a specific attribute it is difficult to express the presence of water control measures in an expert system. Depth to water table might reflect it but not necessarily since this factor is meant to reflect availability of water in the dry season although it was actually considered first and determined to be <3, this does'nt add anything. This relationship really requires an explicit factor as expressed in the PROLOG system, but even if such a factor had been included depending on the adequacy of the data set the priority assumption would probably have precluded it's assessment. It can be concluded from this that more care must be exercised in building the expert system by adjusting attribute value classes where possible and expanding the data set. The extent of the expert interference in the induction process would seem to indicate that automatic induction has certainly not been achieved and this particular method actually seems to complicate rather than simplify the land evaluation process to the extent that it becomes an impractical approach.

```
2 PADDY LAND   If 'PADDY SOIL' OR 'ANNUAL FLOODING'
               AND 'FLAT LAND'
               AND NOT 'CURRENT TREECROP'
```

ALLUVIAL series covers the above prequisites quite closely although no account was taken of the presence of a treecrop. All alluvial soils flood annually and no non-alluvial series were considered to flood.

```
3 CASSAVA If 'Land does not flood'
              AND 'NO SOIL EROSION' OR  No Capital
                   (Not covered in Super Expert system.)
              AND 'SLOPE <B'         OR  NO CAPITAL
                   (Not detected.)
              AND 'Cashflow Important' OR  INERTIA
                   (Indicated by NO EXP and CAP=C >CASSAVA
              AND 'Easy Management Required'  OR  INERTIA
                   (Neither factor covered in Super Expert
              AND 'Minimise Risk'
                   (Tenure might have indirectly expressed
                    this but the factor was not used)
              AND 'LABOUR available'  OR  'SIZE'  <5 Rai
                   (Not used)                (Not applied)
              AND 'Not Current Tree crop
                   (No account taken of current crop.)
```

The above rule illustrates the kind of sophisticated rule that is typical of those derived from a farmer interview with conditions not anticipated or readily recorded on a standard interview form. Factors such as 'Easy Management' are not generally made explicit by the farmer except on close questioning as to why he does not grow one crop instead of another. Every situation cannot be anticipated for the interviewer by designing a form to cover every eventuality, instead the interviewer must use his own skill in eliciting the important factors from a farmer. Unfortunately that information is usually lost later as extra comments are not easily entered into a database.

It can be seen that some of the values of important attributes were consistent in their influence with the PROLOG rules but they were not exercised in combination with the other factor values as the expert system has again used the minimum of attributes to differentiate examples from the data set. The indication is again that many more examples are needed.

Other rule matches immediately evident:-

i) The suitability of rubber for larger areas:

From farmer comment a site of 50 rai was recognised as the critical point, in fact the system induced a rule of over 71 rai then RUBBER. Although yet again other important factors such as tenure and agroclimatic were not queried.

ii) The moist agroclimatic division for coffee was detected where no previous experience was recorded or rubber had been grown before i.e.

```
COFFEE  If 'No previous experience' OR  'Rubber'
        AND 'Capital A'
        AND 'Moist agroclimate'
```

This is again a very simplistic rule when compared to that derived from farmer comment.

Table showing Yasothon data used in SuperExpert

LU	WATER	AGE	EDUCATE	DOYRS	JOBNO	TEN1	TENRAI1	CROP
Y5	IRRIGATED	52	B	37	0	A	13	BAS-RICE
Y5	IRRIGATED	46	B	31	0	A	10	BAS-RICE
Y5	IRRIGATED	54	B	40	0	A	5	BAS-RICE
Y5	IRRIGATED	46	B	30	1	A	9	BAS-RICE
Y11	RAINFED	39	B	10	0	A	5	CASSAVA(F)
Y11	RAINFED	64	B	10	0	A	8	CASSAVA(F)
Y11	RAINFED	52	B	15	0	A	20	CASSAVA(F)
Y11	RAINFED	58	B	20	0	A	30	CASSAVA(F)
Y11	RAINFED	40	B	15	0	A	6	CASSAVA(F)
Y11	RAINFED	45	B	5	0	A	20	CASSAVA(F)
Y11	RAINFED	33	B	3	0	A	10	CASSAVA(F)
Y11	RAINFED	60	B	5	0	A	7	CASSAVA(F)
Y11	RAINFED	47	C	15	1	A	10	CASSAVA(N)
Y11	RAINFED	47	B	4	0	A	11	CASSAVA(N)
Y11	RAINFED	39	B	6	0	A	8	CASSAVA(N)
Y11	RAINFED	55	B	25	5	A	7	CASSAVA(N)
Y11	RAINFED	42	B	2	2	A	14	CASSAVA(N)
Y11	RAINFED	63	B	6	0	A	7	CASSAVE(N)
Y11	RAINFED	35	B	3	0	A	40	CASSAVA(N)
Y14	RAINFED	47	C	5	1	A	10	CASSAVA(N)
Y5	RAINFED	43	B	28	2	A	10	MALICHICOR
Y5	RAINFED	33	B	20	0	A	4	MALICHICOR
Y5	RAINFED	37	B	22	0	A	5	MALICHICOR
Y5	RAINFED	54	B	42	0	A	38	MALICHICOR
Y1	RAINFED	50	B	20	0	D	0	CHILLI(F)
Y1	RAINFED	25	B	1	0	D	0	CHILLI(F)
Y1	RAINFED	34	B	1	1	D	0	CHILLI(F)
Y1	RAINFED	40	B	25	0	D	0	CHILLI(F)
Y5	RAINFED	43	B	28	2	A	10	MALICHICOR
Y5	RAINFED	59	B	34	0	A	14	MALICHICOR
Y5	RAINFED	34	B	20	3	A	7	MALICHICOR
Y1	RAINFED	31	B	2	0	D	1	SWEETCORN
Y1	RAINFED	40	B	2	0	D	0	SWEETCORN
Y1	RAINFED	57	B	2	0	D	0	SWEETCORN
Y1	RAINFED	47	B	3	0	D	0	SWEETCORN
Y1	RAINFED	37	B	1	0	D	0	SWEETCORN
Y1	RAINFED	57	B	2	0	D	0	SWEETCORN
Y5	RAINFED	51	B	30	0	A	3	GLUT
Y3	RAINFED	39	B	26	0	A	5	GLUT
Y3	RAINFED	53	B	38	0	A	7	GLUT
Y3	RAINFED	56	B	44	1	A	8	GLUT
Y3	RAINFED	46	B	31	1	A	12	GLUT
Y10	RAINFED	30	B	15	0	A	6	GLUT
Y10	RAINFED	51	B	25	0	A	20	GLUT
Y10	RAINFED	51	A	36	0	A	10	GLUT
Y10	RAINFED	64	B	40	0	A	5	GLUT
Y10	RAINFED	60	A	44	0	A	10	GLUT
Y10	RAINFED	65	B	50	0	A	17	GLUT
Y10	RAINFED	38	B	23	0	A	10	GLUT
Y10	RAINFED	41	B	26	1	A	10	GLUT
Y3	RAINFED	46	B	30	0	A	6	GLUT

Y3	RAINFED	39	B	26	0	A	5	GLUT
Y3	RAINFED	53	B	38	0	A	7	GLUT
Y3	RAINFED	56	B	44	1	A	8	GLUT
Y3	RAINFED	46	B	31	1	A	12	GLUT
Y3	RAINFED	47	B	32	0	A	10	GLUT
Y3	RAINFED	55	B	40	0	A	25	GLUT
Y2	IRR+RAIN	57	B	15	2	A	20	GLUT
Y2	IRR+RAIN	36	C	21	0	A	20	GLUT
Y2	IRR+RAIN	49	B	35	1	A	10	GLUT
Y5	IRRIGATED	55	B	40	0	A	3	GLUT
Y5	IRRIGATED	42	B	27	1	A	3	GLUT
Y5	IRRIGATED	45	B	20	0	A	10	GLUT
Y5	IRRIGATED	72	B	57	2	A	5	GLUT
Y11	RAINFED	45	B	30	0	A	5	GROUNDFAK
Y11	RAINFED	45	B	18	0	A	5	GROUNDFAK
Y11	RAINFED	43	B	20	1	A	5	GROUNDFAK
Y11	RAINFED	58	B	4	0	A	5	GROUNDFAK
Y2	IRR+RAIN	57	B	15	2	A	20	GLUTGRD
Y2	IRR+RAIN	36	C	21	0	A	20	GLUTGRD
Y2	IRR+RAIN	49	B	35	1	A	10	GLUTGRD
Y2	IRR+RAIN	29	C	17	0	A	4	GLUTGRD
Y5	RAINFED	43	B	22	2	A	3	MALIGRD
Y5	RAINFED	41	B	21	0	A	10	MALIGRD
Y5	RAINFED	41	B	27	0	A	5	MALIGRD
Y5	RAINFED	49	B	37	2	A	8	MALIGRD
Y11	RAINFED	50	B	10	0	D	2	KENAF
Y11	RAINFED	35	B	20	0	D	2	KENAF
Y11	RAINFED	29	B	5	0	A	2	KENAF
Y11	RAINFED	35	B	15	0	A	4	KENAF
Y11	RAINFED	40	B	3	0	A	4	KENAF
Y11	RAINFED	40	B	11	0	A	12	KENAF
Y11	RAINFED	51	B	20	0	A	4	KENAF
Y11	RAINFED	40	B	10	0	A	5	KENAF
Y11	RAINFED	27	B	5	0	A	10	KENAFMELON
Y11	RAINFED	45	B	9	0	A	14	KENAFMELON
Y11	RAINFED	50	B	25	0	A	6	KENAFMELON
Y11	RAINFED	42	B	12	0	A	15	KENAFMELON
Y11	RAINFED	44	B	10	0	A	5	KENAFMELON
Y11	RAINFED	25	B	5	2	A	5	KENAFMELON
Y11	RAINFED	52	B	6	0	A	5	KENAFMELON
Y11	RAINFED	45	B	27	0	A	13	KENAFMELON
Y14	RAINFED	28	C	5	0	A	7	KENAF
Y14	RAINFED	28	C	13	0	A	6	KENAF
Y14	RAINFED	57	B	15	0	A	6	KENAF
Y14	RAINFED	22	C	6	0	A	15	KENAF
Y11	RAINFED	45	B	30	0	A	5	KENAFMELON
Y11	RAINFED	45	B	18	0	A	5	KENAFMELON
Y11	RAINFED	43	B	20	1	A	5	KENAFMELON
Y11	RAINFED	58	B	4	0	A	5	KENAFMELON
Y11	RAINFED	55	B	25	5	A	10	KENAFMELON
Y11	RAINFED	45	B	10	0	A	5	KENAFMELON
Y11	RAINFED	36	B	2	0	A	5	KENAFMELON
Y14	RAINFED	38	B	6	0		0	KENAFMELON
Y1	RAINFED	35	B	2	1	A	4	KENAFKUB
Y1	RAINFED	58	B	10	0	D	6	KENAFKUB
Y1	RAINFED	56	B	3	0	D	4	KENAFKUB

Y1	RAINFED	30	B	5	0	D	3	KENAFKUB
Y14	RAINFED	28	B	20	0	A	5	MANGO
Y14	RAINFED	59	B	16	0	A	4	MANGO
Y14	RAINFED	38	C	15	0	A	7	MANGO
Y14	RAINFED	44	B	8	0	A	8	MANGO
Y14	RAINFED	40	C	1	0	A	10	MANGO
Y14	RAINFED	38	C	4	0	A	7	MANGO
Y14	RAINFED	28	B	4	0	A	4	MANGO
Y14	RAINFED	44	B	3	0	A	8	MANGO
Y14	RAINFED	40	C	1	0	A	10	MANGO
Y14	RAINFED	70	C	12	0	A	2	MANGO
Y14	RAINFED	38	C	4	0	A	7	MANGO
Y14	RAINFED	26	B	4	0	A	4	MANGO
Y14	RAINFED	44	B	8	0	A	8	MANGO1
Y14	RAINFED	70	C	12	0	A	2	MANGO1
Y14	RAINFED	28	B	20	0	A	5	MANGO2
Y14	RAINFED	59	B	16	0	A	4	MANGO2
Y11	RAINFED	45	B	9	0	A	14	KENAFMELON
Y11	RAINFED	42	B	12	0	A	15	KENAFMELON
Y11	RAINFED	44	B	10	0	A	5	KENAFMELON
Y11	RAINFED	25	B	5	2	A	5	KENAFMELON
Y11	RAINFED	52	B	6	0	A	5	KENAFMELON
Y11	RAINFED	27	B	5	0	A	10	KENAFMELON
Y11	RAINFED	45	B	27	0	A	13	KENAFMELON
Y11	RAINFED	60	B	24	0	A	5	KENAFMELON
Y11	RAINFED	45	B	30	0	A	5	KENAFMELON
Y11	RAINFED	45	B	18	0	A	5	KENAFMELON
Y11	RAINFED	43	B	20	1	A	5	KENAFMELON
Y11	RAINFED	58	B	4	0	A	5	KENAFMELON
Y11	RAINFED	55	B	25	5	A	10	KENAFMELON
Y11	RAINFED	45	B	10	0	A	5	KENAFMELON
Y11	RAINFED	45	B	5	0	A	10	KENAFMELON
Y14	RAINFED	38	B	6	0		0	KENAFMELON
Y5	RAINFED	51	B	36	1	A	2	MALIMEL
Y5	RAINFED	47	B	32	0	A	15	MALIMEL
Y5	RAINFED	40	B	22	0	A	3	MALIMEL
Y5	RAINFED	27	B	17	0	A	18	MALIMEL
Y1	RAINFED	63	B	7	0	D	2	NUT
Y1	RAINFED	67	B	5	0	A	2	NUT
Y1	RAINFED	56	B	10	0	A	1	NUT
Y1	RAINFED	67	B	3	0	D	1	NUT
Y5	RAINFED	43	B	28	2	A	10	MALICHICOR
Y5	RAINFED	33	B	20	2	A	4	MALICHICOR
Y5	RAINFED	37	B	22	0	A	5	MALICHICOR
Y5	RAINFED	54	B	42	0	A	38	MALICHICOR
Y5	IRRIGATED	52	B	37	0	A	13	BAS-RICE
Y5	IRRIGATED	46	B	31	0	A	10	BAS-RICE
Y5	IRRIGATED	54	B	40	0	A	5	BAS-RICE
Y5	IRRIGATED	46	B	30	1	A	9	BAS-RICE
Y5	IRRIGATED	55	B	40	0	A	3	GLUTRICE
Y5	IRRIGATED	45	B	20	0	A	10	GLUTRICE
Y5	IRRIGATED	72	B	57	2	A	5	GLUTRICE
Y5	IRRIGATED	48	B	36	0	A	20	GLUTRICE
Y10	RAINFED	30	B	9	0	A	6	MALIRICE
Y10	RAINFED	43	C	28	0	A	10	MALIRICE
Y10	RAINFED	64	C	26	0	A	10	MALIRICE

Y10	RAINFED	51	A	31	0	A	9	MALIRICE
Y10	RAINFED	60	A	44	0	A	10	MALIRICE
Y10	RAINFED	38	B	23	0	A	8	MALIRICE
Y10	RAINFED	41	B	25	1	A	4	MALIRICE
Y3	RAINFED	59	B	43	1	A	20	MALIRICE
Y3	RAINFED	61	B	46	1	A	12	MALIRICE
Y3	RAINFED	55	B	40	0	A	18	MALIRICE
Y3	RAINFED	46	B	30	0	A	6	MALIRICE
Y3	RAINFED	50	B	35	0	A	12	MALIRICE
Y3	RAINFED	54	B	29	0	A	20	MALIRICE
Y3	RAINFED	43	B	28	0	A	20	MALIRICE
Y3	RAINFED	61	B	46	0	A	4	MALIRICE
Y4	RAINFED	41	B	26	0	A	4	MALIRICE
Y4	RAINFED	58	B	43	0	A	7	MALIRICE
Y4	RAINFED	60	B	50	0	A	22	MALIRICE
Y4	RAINFED	60	B	30	0	A	30	MALIRICE
Y4	RAINFED	41	B	5	0	A	16	MALIRICE
Y5	RAINFED	33	B	2	0	A	8	MALIMEL
Y5	RAINFED	47	B	32	0	A	15	MALIMEL
Y5	RAINFED	40	B	22	0	A	3	MALIMEL
Y5	RAINFED	27	B	17	0	A	18	MALIMEL

YASEXAM.PRB

Example [illegible]

of [illegible]

```
perExpert: Rule listing
oblem: /CROP

LU
  YAS5 : WATER
     RAINFED : TENRAI1
          <  4 : JOBNO
               <  2 : AGE
                    <  46 : MALIMEL
                    >= 46 : DOYRS
                         <  33 : GLUT
                         >= 33 : MALIMEL
               >= 2 : MALIGRD
          >= 4 : DOYRS
               <  19 : MALIMEL
               >= 19 : TENRAI1
                    <  15 : AGE
                         <  39 : MALICHICOR
                         >= 39 : AGE
                              <  42 : MALIGRD
                              >= 42 : DOYRS
                                   <  36 : MALICHICOR
                                   >= 36 : MALIGRD
                    >= 15 : AGE
                         <  51 : MALIMEL
                         >= 51 : MALICHICOR
     IRR-RAIN : NULL
          IRRIGATED : AGE
               <  55 : AGE
                    <  46 : AGE
                         <  44 : GLUT
                         >= 44 : CLASH-GLUT-GLUTRICE
                         >= 46 : TENRAI1
                              <  17 : BAS-RICE
                              >= 17 : GLUTRICE
                    >= 55 : AGE
                         <  64 : CLASH-GLUTRICE-GLUT
                              >= 64 : CLASH-GLUTRICE-GLUT
                         IRR+RAIN : NULL
                         YAS1 : TEN1
                              A : EDUCAT
                                   A : DOYRS
                                         <  34 : MALIRICE
```

```
 >= 34 : AGE
     <  56 : GLUT
     >= 56 : CLASH-MALIRICE-GLUT
B : TENRAT1
     <  5 : AGE
         <  54 : AGE
             <  29 : MANGO
             >= 29 : JOBNO
                  <  1 : KENAF
                  >= 1 : AGE
                      <  38 : KENAFKUB
                      >= 38 : MALIRICE
         >= 54 : DOYRS
             <  13 : NUT
             >= 13 : CLASH-MANGO-MANGO2
         >= 5 : TENRAT1
             <  6 : AGE
                  <  42 : DOYRS
                      <  8 : KENAFMELON
                      >= 8 : AGE
                          <  34 : CLASH-MANGO-MANGO2
                              >= 34 : AGE
                                  <  40 : CASSAVAF
                                  >= 40 : KENAF
                  >= 42 : AGE
                      <  62 : DOYRS
                          <  14 : AGE
                              <  55 : KENAFMELON
                              >= 55 : CLASH-KENAFMELON-GROUNDFAK
                              >= 14 : AGE
                                  <  53 : AGE
                                      <  44 : CLASH-KENAFMELON-GROUNDFAK
                                          >= 44 : DOYRS
                                              <  24 : CLASH-KENAFMELON-GROUNDFAK
                                                  >= 24 : CLASH-KENAFMELON-GROUNDFAK
                                              >= 53 : KENAFMELON
                                      >= 62 : GLUT
                              >= 6 : DOYRS
```

```
< 9 : TENRAT1
  < 9 : DOYRS
    < 7 : DOYRS
      < 4 : MANGO
      >= 4 : DOYRS
        < 6 : CASSAVAF
        >= 6 : CASSAVAN
    >= 7 : CLASH-MANGO-MANGO1
    >= 9 : DOYRS
      < 5 : AGE
        < 34 : CASSAVAF
        >= 34 : CASSAVAN
      >= 5 : TENRAT1
        < 15 : KENAFMELON
        >= 15 : CASSAVAF
  >= 9 : AGE
    < 42 : AGE
      < 39 : DOYRS
        < 12 : MALIRICE
        >= 12 : AGE
          < 34 : GLUT
          >= 34 : TENRAT1
            < 9 : MALIRICE
            >= 9 : GLUT
      >= 39 : AGE
        < 41 : DOYRS
          < 13 : KENAF
          >= 13 : CASSAVAF
        >= 41 : GLUT
    >= 42 : AGE
      < 56 : TENRAT1
        < 18 : TENRAT1
          < 9 : AGE
            < 53 : KENAFMELON
            >= 53 : CASSAVAN
          >= 9 : KENAFMELON
        >= 18 : AGE
          < 52 : GLUT
          >= 52 : CASSAVAF
      >= 56 : DOYRS
        < 30 : AGE
          < 58 : KENAF
```

```
                                      >= 58 : CASSAVAF
                                 >= 30 : AGE
                                      <  65 : MALIRICE
                                      >= 65 : GLUT
   C : AGE
         <  42 : AGE
              <  33 : KENAF
              >= 33 : MANGO
         >= 42 : AGE
              <  59 : AGE
                   <  45 : MALIRICE
                   >= 45 : CASSAVAN
               >= 59 : CLASH-MANGO1-MANGO
   D : TENRAT1
          <  1 : DOYRS
               <  12 : DOYRS
                    <  2 : AGE
                         <  36 : CHILLIF
                         >= 36 : SWEETCORN
                    >= 2 : SWEETCORN
               >= 12 : CHILLIF
          >= 1 : TENRAT1
                <  3 : AGE
                     <  57 : KENAF
                     >= 57 : NUT
                >= 3 : KENAFKUB
YAS2 : AGE
       <  33 : GLUTGRD
       >= 33 : AGE
            <  43 : CLASH-GLUTGRD-GLUT
                >= 43 : AGE
                     <  53 : CLASH-GLUTGRD-GLUT
                          >= 53 : CLASH-GLUTGRD-GLUT
               YAS3 : AGE
                      <  57 : TENRAT1
                           <  15 : AGE
                                <  52 : AGE
                                     <  49 : DOYRS
                                         <  31 : AGE
                                               <  43 : GLUT
                                               >= 43 : CLASH-MALIRICE-GLUT
                                               >= 31 : GLUT
                                          >= 49 : MALIRICE
                                 >= 52 : GLUT
                            >= 15 : TENRAT1
                                 <  23 : MALIRICE
                                 >= 23 : GLUT
                       >= 57 : MALIRICE
                 YAS4 : MALIRICE
                 YAS10 : NULL
                      YAS11 : NULL
                           YAS13 : NULL
                                YAS14 : NULL
```

Example of the Soil Survey

SERIAL NUMBER	003743

PROJECT				OBS TYPE	DATE						DESCRIPTION				LOCATION								GRID REFERENCE							
					DAY		MONTH		YEAR		AUTHOR		CHECKER		LINE				POINT				E				N			

OBS DEPTH (cm)		DEPTH TO (cm)				DEPTH OF (cm)				CLASSIFICATION						TEXT
		IMP LAYER		WATER TABLE		LITTER		ROOTING		PROVISIONAL			FINAL			

ELEVATION (metres)				POSITION		LAND FORM	SLOPE°		ASPECT			ROCK FORM	Parent Mater'l	Drainage	PEAT				VEGETATION TYPE		CROPS				MANAGEMENT					Erosion			TEXT
				I	II										Type	DEPTH					ANNUAL		PERENNIAL		Tech level	Systems power	Systems adap	Tenure	Field size ha				
0	0	0	0	Crest	T1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				0
1	1	1	1	Upp	T2	Flat	1	1	N	N	N	I	Resid	1	Fibric Forest	1	1	1	1	1	1	1	1	1	+fym	hand	shift cult	lease	<1/10	Sheet			1
2	2	2	2	Mid	Back Swamp	Und	2	2	E	E	E	II	Coll	2	Sapric Forest	2	2	2	2	2	2	2	2	2	+fert	animal	bush fallow	Custom us fruct	1/10 1	Gully			2
3	3	3	3	Low	Dpn	Rolling	3	3	S	S	S	III	Fluvial	3	Haemic Forest	3	3	3	3	3	3	3	3	3	+fym +fert	Inter wheel	perm. rainfed	tenant	1 10	Rill			3
4	4	4	4	Val Bot	Ridge	Hilly	4	4	W	W	W	IV	Aeoln	4	Fibric Grass	4	4	4	4	4	4	4	4	4	+cults	wheel	perm irrigd	share	10 100	Wind			4
5	5	5	5	Plain	Swale	Diss	5	5				V	Marine	5	Sapric Grass	5	5	5	5	5	5	5	5	5	+ferts +cults	track	perm inund	co op	>100				5
6	6	6	6	Flat	Levee	Mount	6	6				VI,	Lacus	6	Haemic Grass	6	6	6	6	6	6	6	6	6				land sett.					6
7	7	7	7	Terr	R. bed		7	7				VII		7		7	7	7	7	7	7	7	7	7				free hold					7
8	8	8	8		Other	Other	8	8				Other		8	Other	8	8	8	8	8	8	8	8	8	Other	Other	Other	Other		Other			8
9	9	9	9	NR	NR	NR	9	9	NR	NR	NR	NR	NR	NR	NR	9	9	9	NR	9	NR	9	NR	9	NR	NR	NR	NR	NR	NR			9

HOR	DEPTH (cm)			MAIN COLOUR				MOTTLE 1						MOTTLE 2						TEXTURE			CON SIS		INCLUSIONS						Sal	pH	TEXT
								%	m.m	°Con	COLOUR			%	m.m.	°Con	COLOUR						m	d	%	m.m.	type	%	m.m.	type			
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	L	L	0	0	0	0	0	0	0	0	0
	1	1	1	2.5	Y	1	1	< 2	< 5	Ft	pale	gyh	Grey	< 2	< 5	Ft	pale	gyh	Grey	By	C	C	V fri	S	<5	2 20	R1	< 5	2 20	R1	1	1	1
	2	2	2	5	YR	2	2	2 10	5 15	Di	light	ylh	Yellow	2 10	5 15	Di	light	ylh	Yellow	St	Sl	Sl	fri	SH	5 15	20 75	R2	5 15	20 75	R2	2	2	2
	3	3	3	7.5	R	3	3	10 20	> 15	Pr	dark	brh	Brown	10 20	> 15	Pr	dark	brh	Brown	Grav	L	L	Fi	H	15 40	75 250	R3	15 40	75 250	R3	3	3	3
	4	4	4	10	GY	4	4	20 30			Strong	rdh	Red	20 30			Strong	rdh	Red	Grit	VFS	VFS	VFi	VH	40 80	>250	R4	40 80	> 250	R4	4	4	4
	5	5	5		G	5	5	30 40				bluh	White	30 40				bluh	White	Fy	FS	FS	XF	XH			C1			C1		5	5
	6	6	6		BG	6	6	> 40				grnh	Green	> 40				grnh	Green	My	CS	CS	unripe				C2			C2		6	6
	7	7	7		B	7	7	mix					Black	mix					Black	Bst	S	S					C3			C3		7	7
	8	8	8			8	8															CL										8	8
	9	9	9	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	NR	9
1																																	
2																																	
3																																	
4																																	
5																																	
6																																	

APPENDIX C

System Structure
Rule Listing
Accuracy Assessment

```
EXERCISE SESSION FOR THE FOLLOWING EXPERT SYSTEMS:
--------------------------------------------------
       FARMER'S DECISION

Test cases generated manually.

Choices not narrowed down.

   Does this individual have long term tenure of his farm holding either
by owning the freehold or by a permanent tennancy agreement ?
Answer: YES

   What savings does the farmer have available as working capital for
a new crop venture?
     Description                                              Category
Affluent, no borrowings                                          1
borrowings but wealthy                                           2
Borrowings and low income                                        3
Heavily indebted and low income                                  4
Answer: 2

   Does the farmer have any income from other employment such as
labouring work ? Bussiness interests are not intended to be covered by
this question. Please answer Yes or No
Answer: YES

   Is the farmer engaged in any other business ventures which supplement
his farm income ?
Answer: NO

   Does the farmer have a long term outlook or is he only interested in
making money on a year by year basis ? Answer yes if you think the farmer
is prepared to consider investment return over at least a ten year period.
Answer: YES

   Is there a shortage of labour in the area ?
Answer: NO

   What is the distance to the nearest sugar factory ? Please answer to
the nearest kilometre
Answer: 28

   What is the distance to the nearest pineapple factory ? Please answer
to the nearest kilometre.
Answer: 60

   How large is the farm holding ? Please enter the approximate area in Rai.
Answer: 21

Select the appropriate slope class from the table below.
    PERCENTAGE SLOPE                              CATEGORY
        0<2                                          A
        2<5                                          B
        5<8                                          C
        8<16                                         D
       16<35                                         E
         >35                                        SC
Answer: B
```

```
 Enter the approximate depth to the water table at the height of the
dry season in cm.
Answer: 450

 Does the land normally flood each year
Answer: NO

Enter the code No for the appropriate USDA soil class.
Typic Quartzi Psamments  1       Orthoxic Dystropepts  6
Typic Paleaquults        2       Aquic Arenic Eutrochrepts       7
Typic paleudults         3       Typic Tropudults                8
Rhodic paleudults        4
Dystropeptic Tropudults  5
Answer: 1

CROP CHOICE:

Para rubber is recommended
(44)

CASSAVA
(19)

COCONUT
(19)

DURIAN
(8)

Groundnuts are recommended if appropriate to the farming system and
suitable for the land.
(8)

Reliability = 42

Another ( Y, N, H, E, or A ) ? N
```

Table C.1 showing the accuracy with which the TIMM system predicted the farmer's crop choice.

No.	Suggested Crops (% Certainty)	Suggested Reliability	Actual Crops (% Area)

1	Rubber(77)
	Cassava(17)
	Coconut(3)
	Durian(1)
	Groundnuts(1)
		62	Rubber(75)
			Durian(16)
			Rambutan(9)
			[+Cassava & groundnut as intercrops]

2	Durian(66)
	Coffee(33)
	Rubber(0)
	Cassava(0)
	Coconut(0)
			Durian(84)
			Rambutan(16)

3	Durian(25)
	Cassava(22)
	Coconut(22)
	Pineapple(17)
	Rubber(11)
		52	Rubber(50)
			Durian(20)
			Rice(5)

4	Rubber(44)
	Cassava(18)
	Coconut(18)
	Durian(10)
	Groundnuts(8)
	Coffee(1)
		60	Sugar cane(86)
			Rubber(13)
			Durian(1)

5	Durian(37)
	Rubber(26)
	Coconut(17)
	Cassava(17)
	Groundnuts(3)
		54	Cassava

6	Rubber(45)
	Cassava(19)
	Coconut(19)
	Durian(8)
	Groundnuts(8)
		53	Rubber
			Bananas
			Some Cassava intercropping

7	Cassava(50)
	Coconut(50)
		63	Coconut(9Rai)
			Cassava(7)
			Paddy(2)

8	Rubber(45)
	Cassava(19)
	Coconut(19)
	Durian(8)
	Groundnuts(8)
			Cassava

9	Paddy rice(100)
	[Recommendation ignored the]
	[possibility for mounding]
		63	Coconut(7)
			Paddy(3)
			Mangosteen(8)
			Cassava as intercrop

10	Rubber(81)
	Cassava(17)
	Coconut(1)
	Durian(0)
	Groundnuts(0)
		62	Coconut(4)
			Cassava(7)
			Paddy(20)
			Durian(4)
			Pineapple(5)

11	Cassava(50)
	Coconut(50)
		87	Mangosteen(3)
			Durian(3)
			Cassava(2)

12	Pineapple(30)
	Sugarcane(30)
	Rubber(18)
	Coconut(8)
	Cassava(8) > [Farmer said land was too infertile]
	Groundnuts(3)
	Durian(3) > [Farmer said land was unsuitable]
		70	Jubjub(4)

13	Sugarcane(100)	80	Lamud(4)
			Coconut(2)
			Beans

	[The farmer commented that his sandy soil was suitable for La-mud; mango; sweet apple; or coconut.		

14	Cassava(33)
	Coconut(33)
	Rubber(33)
		80	Cassava(10)
			Coconut(1)
			Rice(6)
			Durian(2)

15	Durian(63)
	Coconut(12)
	Coffee(12)
	Rubber(12)
		66	Cassava(24)

	[The farmer said his land was too 'high' (dry) for coconut or durian. He would have grown rubber if he had enough money to invest.]		

16	Cassava(29)
	Coconut(29)
	Rubber(17)
	Durian(12)
	Groundnuts(12)
		72	Cassava(7)
			Pineapple(4)
			Coconut(4)

	[The farmer intended to grow rubber in the future, but commented there was insufficient water available for durian.]		

17	Durian(65)
	Coffee(33)
	Coconut(1)
	Rubber(1)
		81	Cassava(30)
			Coconut(30)

18	Rubber(75)
	Cassava(15)
	Coconut(6)
	Coffee(2)
	Durian(2)
		72	Cassava(10)
			Pineapple(10)
			Jackfruit(10)
			Coconut(8)

19	Durian(100)	73	Eucalyptus(164)
			Pinus(50)
			Sugar cane(35)
			Cassava(147)
			Pineapple(20)

20	Rubber(28)
	Durian(24)
	Coconut(20)
	Coffee(20)
	Cassava(7)
			Rubber(40)
			Pineapple(7)
			Durian(3)

[The farmer commented that his land was too dry for coffee]

22	Rice(94)
	Durian(6)
		64	Rice

[Durian was recommended even though the land flooded annually]

23	Vegetables(100)	61	Cashew Sathorn Mango Jackfruit

[Vegetables were recommended because of the soil type even though the water table was at least 6m from the surface making irrigation uneconomic]
The farm was actually located on an association of soils. When the soil type was given to the TIMM system as the minor soil in the association (Typic paleudults rather than Orthoxic dystropepts) the recommendation was quite different, because there were many more examples of this soil type.
Viz: -> Cassava(33)
Coconut(33)
Rubber(33)
This would be a reasonable recommendation, although not apparently correct in the farmer's view.

26	Rubber(28)
	Durian(24)
	Coffee(20)
	Coconut(20)
	Cassava(7)
		76	Durian(2%)
			Mangosteen(6%)
			Sapodilla(8%)
			Rubber(84%)

27	Rubber(37)
	Banana(37)
	Cassava(25)
		57	Rubber(14)
			Cassava with young rubber(30)

28	Rubber(100)	67	Durian(7)
			Cassava(10)

29	Rubber(28)
	Durian(24)
	Coffee(20)
	Coconut(20)
	Cassava(7)
		76	Durian(20)

APPENDIX D

Program Listing
Database Structures
Screen dumps of interactive sessions

```
Structure for database : a:\eval\landinf.dbf
Number of data records :       10
Date of last update    : 01/01/80
Field  Field name  Type       Width    Dec
    1  RECORD      Numeric        3
    2  LU          Character      3
    3  CODE        Character     10
    4  SOILNAME    Character     22
    5  DRAIN       Character      2
    6  SLOPE       Character      2
    7  SURFTEXT    Character      2
    8  SUBSTEXT    Character      2
    9  DEPTH       Character      3
   10  AGROCLIM    Character      2
   11  CECSF       Character      2
   12  CECSB       Character      2
   13  SURPH20     Character      2
   14  SUBPH20     Character      2
   15  KSF         Character      2
   16  PSF         Character      2
** Total **                      64
```

```
Structure for database : a:\eval\cropsuit.dbf
Number of data records :       32
Date of last update    : 02/02/88
Field  Field name  Type       Width    Dec
    1  RECORD      Numeric        3
    2  CROP        Character     12
    3  RATING      Character      3
    4  DRAIN       Character     10
    5  SLOPE       Character      5
    6  SURFTEXT    Character     12
    7  SUBSTEXT    Character     12
    8  DEPTH       Character     10
    9  AGROCLIM    Character      2
   10  CECSF       Character      7
   11  CECSB       Character      7
   12  SURPH20     Character      7
   13  SUBPH20     Character      7
   14  KSF         Character      5
   15  PSF         Character      7
** Total **                     110
```

```
Structure for database : a:\eval\factors.dbf
Number of data records :       12
Date of last update    : 01/01/80
Field  Field name  Type       Width    Dec
    1  QUALITY     Character     10
** Total **                      11
```

```
Structure for database : a:\eval\selectio.dbf
Number of data records :        1
Date of last update    : 01/01/80
Field  Field name  Type       Width    Dec
    1  CHOICE      Character     12
    2  SELECT      Character      4
** Total **                      17
```

```
Structure for database : a:\eval\currsuit.dbf
Number of data records :        1
Date of last update    : 01/01/80
Field  Field name  Type       Width    Dec
    1  CROP        Character     12
    2  LU          Character      3
    3  RATING      Numeric        2
** Total **                      18
```

```
                    LAND SUITABILITY EVALUATION

                              OPTIONS
                              -------
         Select your next option and enter the number of your choice

  •      1        An evaluation for selected crop & land unit

         2        Check on the Present Value of benefits for the chosen cro

         3        Supply information on a new land unit for evaluation

         4        Add a new crop or improve existing information on limits

         5        Return to display menu

         6        Quit the evaluation system

                                 **********


                    Option ?


         Crops and Land units for which information is presently held are
listed below. Please choose a crop and select a land unit for which you
would like to run the evaluation.

                 CROP                        LAND UNIT
                BANANA*                          1
                CASSAVA*                         2
                COCONUT                          3
                COFFEE                           4
                DURIAN                           5
                GROUNDNUT*                       6
                JUBJUB*                          7
                MANGOSTEEN*                      8
                PINEAPPLE*                       9
                RAMBUTAN*
                RICE
                RUBBER
                SUGARCANE*
                VEGATABLES*

CROP CHOICE DURIAN                LAND UNIT SELECTION 4
```

```
Do you wish to make another selection or choice ?

                    LAND SUITABILITY EVALUATION

          The evaluation for the selections you made is now complete, how
 would you like the results displayed ?

          1.       On the Screen

          2.       On the screen together with a printout

          3.       Only by printout


     Enter the number of your chosen option ?




                              EVALUATION RESULTS


          For  DURIAN             land unit  4

          the rating is   S 4


                                        --------

Press the space bar for the next rating
```

```
**EVALUATE.PRG
PUBLIC ALLC, ALL, FIELD, ASKCROP
SET PROCEDURE TO CHANGES
SET SAFETY OFF
SET EXACT OFF
SET TALK OFF
SELE 1
USE FACTORS ALIAS FACT
SELE 2
USE LANDINF INDEX INFLU ALIAS LAND
GO TOP
SELE 3
USE SELECTION ALIAS SEL
SELE 4
USE CROPSUIT INDEX CROPRAT ALIAS REQ
SELE 3
DO OPTIONS
RETURN
```

```
***Changes.prg, containing procedure files to accomplish a
***suitability classification of land for agriculture
***following FAO principles.
PROCEDURE MATCHER *To match land qualities to requirements
of the chosen crop.
FINAL=.F.
CRATING=1
SELE 4
DO WHILE .NOT. FINAL
IF AT(TRIM(LAND->&FIELD),REQ->&FIELD)<>0
DO NEWFIELD WITH FINAL,CRATING
ELSE
IF TRIM(RATING)<>'4'
SKIP
IF CRATING<VAL(REQ->RATING)
CRATING=VAL(REQ->RATING)
ENDIF
ELSE
?'ERROR, no match'
CANCEL
ENDIF
ENDIF
ENDDO
RETURN
**
PROCEDURE NEWFIELD *Moves onto the next land quality for
consideration
PARAMETER FINAL, CRATING
SELE 1
SKIP
IF .NOT. EOF()
FIELD=QUALITY
SELE 4
DO FINDER
ELSE
DO FINAL WITH FINAL,CRATING
ENDIF
RETURN
**
PROCEDURE FINAL *To update the crop suitability for the
current land unit
PARAMETERS FINAL, CRATING
GO TOP
FIELD=QUALITY
FINAL=.T.
SELE 5
USE CURRSUIT INDEX CURS ALIAS CURR
APPEND BLANK
REPLACE RATING WITH CRATING, CROP WITH SEL->CHOICE, LU WITH
LAND->LU
SELE 4
GO TOP
DO FINDER
RETURN
**
PROCEDURE FINDER*Finds crop record of requirements for S1
rating
GO TOP
SELE 4
SEEK TRIM(SEL->CHOICE)
RETURN
```

```
**
PROCEDURE CAL
IF AT('ALL',CHOICE)=0
ALLC= .F.
SELE 4
DO FINDER
ELSE
SELE 4
GO TOP
ALLC= .T.
ENDIF
SELE 2
IF AT('ALL',SEL->SELECT)<>0
GO TOP
ALL= .T.
*Set verification that all land units are to be considered
ELSE
ALL= .F.
*If selection is not for all land units
SEEK TRIM(SEL->SELECT)
*Find the record giving information on the selected LU
ENDIF
RETURN
**
PROCEDURE STANDARD
DO WHILE .NOT. EOF()
DO CAL
SELE 1
GO TOP
FIELD=QUALITY
SELE 4
IF ALLC=.T.
DO ALLCR WITH FIELD
ENDIF
IF ALL=.T.
DO ALLUS
ELSE
DO MATCHER
ENDIF
SELE 3
SKIP
ENDDO
DO DISPLAY
RETURN
```

```
**
PROCEDURE ALLCR
PARAMETER FIELD
DO WHILE ALLC
IF ALL=.T.
DO ALLUS WITH FIELD
ELSE
DO MATCHER WITH FIELD
ENDIF
SKIP
IF .NOT. EOF()
SELE 4
FIND CURR->CROP
ELSE
ALLC=.F.
ENDIF
ENDDO
RETURN
**
PROCEDURE ALLUS
SELE 2
N=1
DO WHILE N<10
SELE 4
DO MATCHER
SELE 2
SKIP
N=N+1
ENDDO
ALL=.F.
RETURN
**
PROCEDURE RESULTS *Displays answers from evaluation on the
screen
CLEAR
SELE 5
USE CURRSUIT INDEX CURS
@ 2,28 SAY "EVALUATION RESULTS"
DO WHILE .NOT. EOF()
@ 7,9 SAY "For"
@ 7,14 SAY CROP
@ 7,33 SAY "land unit"
@ 7,44 SAY LU
@ 9,9 SAY "the rating is"
@ 9,25 SAY "S"
@ 9,26 SAY RATING
@ 14,33 SAY "________"
@ 16,10 SAY ""
WAIT 'Press the space bar for the next rating'
SKIP
CLEAR
ENDDO
@ 5,17 SAY "That completes the display of evaluation
results"
@ 10,10 SAY ""
WAIT 'Press any key to return to the options menu'
CLEAR
RETURN
```

```
**
PROCEDURE RSLTSPRN *To print out Land Evaluation results
GO TOP
SELE 5
IF AT('A',UPPER(CHOICE))<>1
DO SELECTED
ELSE
REPORT FORM ANSWERS NOEJECT TO PRINT
RETURN
**
PROCEDURE REQUEST*To allow user to select the crop and Land
unit
CLEAR
@ 1,25 SAY "___________________________"
@ 2,24 SAY "¦                 ¦"
@ 3,24 SAY "¦LAND SUITABILITY EVALUATION ¦"
@ 4,24 SAY "¦____________________________¦"
@ 7,1 SAY "This is an interactive program for land
suitability eva"
@ 7,56 SAY "luation, to determine"
@ 8,1 SAY "a rating for a chosen crop on a selected land
unit."
@ 10,1 SAY "Principles for land evaluation as recommended by
the FA"
@ 10,56 SAY "O are followed"
@ 11,1 SAY "throughout (FAO Soils Bulletins No32 and No52)."
@ 13,1 SAY "The present system is a small prototype
developed for R"
@ 13,56 SAY "ayong province to"
@ 14,1 SAY "illustrate how such a system can be used either
indepen"
@ 14,56 SAY "dently or as a"
@ 15,1 SAY "part of a larger GIS."
@ 16,1 SAY ""
WAIT
CLEAR
@ 1,9 SAY "Would you like further background information on
the sy"
@ 1,64 SAY "stem?"
WAIT ' YES or NO' TO LIB
@ 4,9 SAY "Or are You ready to run an evaluation?"
@ 5,9 SAY "(If No I will assume you want to quit already!)"
WAIT ' YES or NO' TO EVAL
CLEAR
IF AT('Y',UPPER(LIB))=1
DO LIBRARY
ENDIF
IF AT('Y',UPPER(EVAL))=0
USE
QUIT
ENDIF
C='Y'
DO WHILE AT('Y',UPPER(C))<>0
APPEND BLANK
@ 2,9 SAY "Crops and Land units for which information is
presently"
@ 2,65 SAY "held are"
@ 3,1 SAY "listed below. Please choose a crop and select a
land un"
@ 3,56 SAY "it for which you"
@ 4,1 SAY "would like to run the evaluation."
@ 6,17 SAY "CROP           LAND UNIT"
@ 7,15 SAY "BANANA*              1"
@ 8,15 SAY "CASSAVA*             2"
@ 9,15 SAY "COCONUT              3"
@ 10,15 SAY "COFFEE              4"
@ 11,15 SAY "DURIAN              5"
@ 12,15 SAY "GROUNDNUT*             6"
@ 13,15 SAY "JUBJUB*                7"
@ 14,15 SAY "MANGOSTEEN*            8"
@ 15,15 SAY "PINEAPPLE*             9"
@ 16,15 SAY "RAMBUTAN*"
@ 17,15 SAY "RICE"
@ 18,15 SAY "RUBBER"
@ 19,15 SAY "SUGARCANE*"
@ 20,15 SAY "VEGATABLES*"
@ 22,1 SAY "CROP CHOICE"
@ 22,13 GET CHOICE PICT "@!"
@ 22,33 SAY "LAND UNIT SELECTION"
@ 22,53 GET SELECT PICT "9NN"
@ 24,1 SAY "You may enter ALL if you wish for your choice or
sele"
@ 24,54 SAY "ction."
READ
@24,1 CLEAR
WAIT "Do you wish to make another selection or choice ?" TO
C
CLEAR
ENDDO
GO TOP
@ 10,30 SAY 'Please Wait, WORKING'
RETURN
**
PROCEDURE DISPLAY*For display of evaluation results
CLEAR
@ 1,21 SAY "LAND SUITABILITY EVALUATION"
@ 3,9 SAY "The evaluation for the selections you made is now
compl"
@ 3,64 SAY "ete, how"
@ 4,1 SAY "would you like the results displayed ?"
@ 6,9 SAY "1.      On the Screen"
@ 8,9 SAY "2.      On the screen together with a printout"
@ 10,9 SAY "3.       Only by printout"
@ 14,9 SAY ""
WAIT '     Enter the number of your chosen option ?' TO N
CLEAR
DO CASE
CASE N='1'
DO RESULTS
DO OPTIONS
CASE N='2'
DO RESULTS
DO RSLTSPRN
CASE N='3'
DO RSLTSPRN
ENDCASE
CLEAR
@ 5,9 SAY "EVALUATION & DISPLAY OF RESULTS COMPLETED"
@ 7,9 SAY "For further options press C, or any other"
@ 8,9 SAY "key to quit this system."
@ 12,1 SAY ""
WAIT "Please make your choice...." TO K
IF UPPER(K)='C'
DO OPTIONS
ENDIF
SELE 5
USE CURRSUIT INDEX CURS
ZAP
SELE 3
USE SELECTION ALIAS SEL
ZAP
CLEAR ALL
**
PROCEDURE CHECKPV *Ascertains Present Value in Baht/rai for
```

```
selected crops
SELE 5
USE CURRSUIT INDEX CURS ALIAS CURR
GO TOP
*DO WHILE .NOT. EOF() -COMMENTED OUT TEMPORARILY FOR
TRANSFER OF 1 ONLY
SELE 6
USE LOTMES ALIAS MESS
ZAP
APPEND BLANK
MCROPLU = TRIM(CURR->CROP)
REPLACE CROPLU WITH "/fcceLOTUS\"+
SUBSTR(MCROPLU,1,6)+TRIM'CURR->LU)+'.WK1~'
APPEND BLANK
REPLACE CROPLU WITH TRIM(CURR->CROP)
*SELE 5
*SKIP
*ENDDO
*SELE 6
*APPEND BLANK
*REPLACE CROPLU WITH '{goto}V3~'
*APPEND BLANK
*REPLACE CROPLU WITH '/rndCROP~'
*APPEND BLANK
*REPLACE CROPLU WITH '/rncCROP~~'
*APPEND BLANK
*REPLACE CROPLU WITH '/xmTITLE~'
COPY TO LOTMES.PRN SDF
ZAP
RETURN
**
PROCEDURE OUTOLOT
RUN PVCHECK.BAT
RETURN
**
PROCEDURE PVDISPLAY.PRG
USE PVANSWER
APPEND FROM DBSMES.PRN SDF
CLEAR
SELE 5
@ 1,17 SAY "RESULTS FROM CALCULATION OF NET PRESENT VALUE"
@ 2,17 SAY "---------------------------------------------"
@ 4,9 SAY "The figure given below represents the average Net
Prese"
@ 4,64 SAY "nt Value for"
@ 5,1 SAY "growing"
@ 5,10 SAY CROP
@ 5,30 SAY "on land unit"
@ 5,43 SAY LU
SELE 6
@ 8,1 SAY "Net Present Value is"
@ 8,23 SAY ANS
@ 13,33 SAY "______________"
@ 16,1 SAY "You can now return to the options menu."
WAIT
CLEAR
ZAP
RETURN
```

```
**
PROCEDURE ALTREQ
SET TALK OFF
SET EXACT OFF
CLEAR
SELE 4
GO TOP
STORE SPACE(15) TO CROPP
STORE SPACE(2) TO C
STORE SPACE(2) TO NEWCR
@ 1,17 SAY "Updating Crop Information"
@ 2,17 SAY "-------------------------"
@ 4,9 SAY "You can now either add a new crop to the
CROPSUIT.DBF o"
@ 4,64 SAY "r"
@ 5,9 SAY "alter existing information in that file for a
particula"
@ 5,64 SAY "r crop."
@ 7,9 SAY "Please indicate your choice below."
@ 9,9 SAY "Add a new crop ?"
@ 9,26 GET NEWCR
@ 11,9 SAY "if NO then choice is assumed to be alteration of
existi"
@ 11,64 SAY "ng limits"
READ
CLEAR
A=.T.
DO WHILE A=.T.
N=1
IF UPPER(NEWCR)="Y"
USE
USE CROPSUIT
@ 2,1 SAY "Which crop do you wish to add?"
@ 2,37 GET CROPP
@ 6,35 SAY "-----------"
READ
CLEAR
@ 2,32 SAY "NEW CROP"
@ 3,32 SAY "--------"
@ 5,9 SAY "For each quality, enter all possible class values
for t"
@ 5,64 SAY "he crop"
@ 6,1 SAY "at each level of suitability i.e. S1,S2,S3 or S4
(equiv"
@ 6,56 SAY "alent to NS)."
@ 8,9 SAY "No commas or spaces are required between each
value."
@ 10,1 SAY "EXAMPLE :        For RUBBER at the S2 level"
@ 12,17 SAY "Soil Surface Texture may be Coarse loamy(2) or"
@ 13,17 SAY "Fine silty(5) or Sandy(1)."
@ 15,17 SAY "This would be recorded as 125 or 251 or 521
etc."
@ 18,9 SAY "It is very important that all possible classes
are cove"
@ 18,64 SAY "red, as"
@ 19,1 SAY "if not a matching error may occur in the
evaluation. In"
@ 19,57 SAY "other words for"
@ 20,1 SAY "a land unit where the missing class occurs
evaluation w"
@ 20,56 SAY "ill give an error."
@ 22,10 SAY ""
WAIT
CLEAR
DO WHILE N<5
APPEND BLANK
REPLACE CROP WITH CROPP
```

```
NR=STR(N)
REPLACE RATING WITH STR(N,1)
DO GETINFO WITH CROPP, N, NEWCR
REPLACE DEPTH WITH LOWER(DEPTH)
N=N+1
ENDDO
USE CROPSUIT ALIAS REQ
INDEX ON CROP+RATING TO CROPRAT
ELSE
@ 4,17 SAY "This routine allows you to change the
information on"
@ 5,1 SAY "crop requirements currently held in the
CROPSUIT.dbf"
@ 7,17 SAY "Information on crucial limits for each quality
should"
@ 8,1 SAY "be supplied for each level of crop rating."
@ 12,35 SAY "----------"
WAIT
CLEAR
@ 2,1 SAY "Which crop are you interested in?"
@ 2,37 GET CROPP
@ 6,35 SAY "----------"
READ
CLEAR
SEEK TRIM(CROPP)
DO WHILE N<5
DO GETINFO WITH CROPP, N, NEWCR
N=N+1
SKIP
ENDDO
GO TOP
ENDIF
@ 2,9 SAY "Changes/Additions completed for the selected
crop, another ?"
WAIT ´Yes or No´ to C
IF UPPER(C)="N"
A=.F.
ENDIF
CLEAR
STORE SPACE(15) TO CROPP
ENDDO
RETURN
**
PROCEDURE GETINFO
PARAMETER CROPP, N, NEWCR
@ 3,2 SAY CROPP
@ 3,25 SAY "RATING S"
@ 3,34 SAY N
@ 5,1 SAY "DRAINAGE CLASS"
@ 5,17 GET DRAIN PICT "9999999999"
@ 5,41 SAY "SLOPE CLASS"
@ 5,54 GET SLOPE PICT "@!"
@ 7,1 SAY "SOIL SURFACE TEXTURE"
@ 7,23 GET SURFTEXT PICT "999999999999"
@ 7,41 SAY "SOIL SUBSURFACE TEXTURE"
@ 7,66 GET SUBSTEXT PICT "999999999999"
@ 9,1 SAY "SOILDEPTH"
@ 9,12 GET DEPTH PICT "NNNNNNNNNN"
@ 9,41 SAY "AGROCLIMATIC DIVISION"
@ 9,64 GET AGROCLIM PICT "99"
@ 11,1 SAY "SURFACE CEC"
@ 11,14 GET CECSF PICT "9999999"
@ 11,41 SAY "SUBSURFACE CEC"
@ 11,57 GET CECSB PICT "9999999"
@ 13,1 SAY "SURFACE pH"
@ 13,13 GET SURPH2O PICT "9999999"
@ 13,41 SAY "SUBSURFACE pH"
@ 13,56 GET SUBPH20 PICT "9999999"
@ 14,1 SAY "(water)"
@ 16,1 SAY "SURFACE POTASSIUM"
@ 16,20 GET KSF PICT "99999"
@ 16,41 SAY "SURFACE PHOSPHATE"
@ 16,60 GET PSF PICT "9999999"
READ
CLEAR
RETURN
**
PROCEDURE OPTIONS
CLEAR
O=0
@ 1,22 SAY "LAND SUITABILITY EVALUATION"
@ 3,31 SAY "OPTIONS"
@ 4,31 SAY "-------"
@ 5,9 SAY "Select your next option and enter the number of
your ch"
@ 5,64 SAY "oice"
@ 7,9 SAY "1       An evaluation for selected crop & land
unit"
@ 9,9 SAY "2       Check on the Present Value of benefits
for the"
@ 9,64 SAY "chosen crop"
@ 11,9 SAY "3        Supply information on a new land unit
for evalu"
@ 11,64 SAY "ation"
@ 13,9 SAY "4        Add a new crop or improve existing
information"
@ 13,64 SAY "on limits"
@ 15,9 SAY "5        Return to display menu"
@ 17,9 SAY "6        Quit the evaluation system"
@ 19,35 SAY "**********"
@ 22,20 SAY "  Option ? " GET O PICT "@Z ##" RANGE 1,6
READ
CLEAR
DO CASE
CASE O=1
SELE 5
USE CURRSUIT INDEX CURS
ZAP
SELE 3
USE SELECTION ALIAS SEL
ZAP
DO REQUEST
DO STANDARD
CASE O=2
DO CHECKPV
DO OUTOLOT
DO PVDISPLAY
DO OPTIONS
CASE O=3
DO USERASK
DO STANDARD
CASE O=4
DO ALTREQ
DO OPTIONS
CASE O=5
DO DISPLAY
CASE O=6
SELE 5
ZAP
SELE 3
ZAP
CLEAR ALL
ENDCASE
RETURN
```

```
**
PROCEDURE USERASK
CLEAR
SELE 2
COPY STRU TO TEMP
USE TEMP
SET TALK OFF
VERIFY="A"
DO WHILE UPPER (VERIFY)="A"
APPEND BLANK
@ 2,9 SAY "DIRECT INPUT OF INFORMATION FOR A NEW LAND UNIT"
@ 4,1 SAY "Please answer the following questions:"
@ 6,1 SAY "What is the code number for this land unit ?"
@ 6,48 GET LU PICT "999"
@ 7,48 SAY "***"
@ 9,1 SAY "Enter the soil code (e.g. Pk for Phuket) ?"
@ 9,46 GET SOILNAME PICT "@A"
@ 11,9 SAY "Choose the appropriate drainage class from the
table be"
@ 11,64 SAY "low"
@ 12,1 SAY "and enter the code number here?"
@ 12,35 GET DRAIN PICT "99"
@ 14,9 SAY "DRAINAGE CLASS                 CODE
NO."
@ 16,9 SAY "Very Poor or Poor                 1"
@ 18,9 SAY "Somewhat poor                     2"
@ 20,9 SAY "Moderately Well                   3"
@ 22,9 SAY "Well                              4"
@ 24,9 SAY "Excessively                       5"
READ
CLEAR
@ 2,25 SAY "TEXTURE CLASSES"
@ 3,25 SAY "---------------"
@ 5,9 SAY "TEXTURE DESCRIPTION           CODE"
@ 7,17 SAY "SANDY                   1"
@ 8,17 SAY "COS LOAMY               2"
@ 9,17 SAY "FINE LOAMY              3"
@ 10,17 SAY "COS SILTY               4"
@ 11,17 SAY "FINE SILTY              5"
@ 12,17 SAY "CLAYEY                  6"
@ 13,17 SAY "SKELETAL                7"
@ 15,9 SAY "From the above table choose the closest
description of"
@ 15,64 SAY "the texture"
@ 16,1 SAY "for each horizon of the soil concerned."
@ 18,9 SAY "Surface Horizons(0-30cm) ?"
@ 18,43 GET SURFTEXT PICT "99"
@ 20,9 SAY "Subsurface Horizons(30-100cm) ?"
@ 20,43 GET SUBSTEXT PICT "99"
READ
CLEAR
@ 1,31 SAY "SOIL DEPTH"
@ 2,31 SAY "----------"
@ 4,5 SAY "Into which soil depth category does the soil
fall?"
@ 4,57 GET DEPTH PICT "NNN"
@ 4,64 SAY "(Enter class)"
@ 6,9 SAY "Description            cm
Class"
@ 8,9 SAY "Very Shallow           0-15
d1"
@ 10,9 SAY "Shallow               15-50
d2"
@ 12,9 SAY "Moderately Deep       50-100
d3"
@ 14,9 SAY "Deep                 100-150
d4"
```

```
@ 16,9 SAY "Very Deep"
@ 16,43 SAY 150
@ 16,59 SAY "d5"
@ 18,32 SAY "********"
READ
CLEAR
@ 1,9 SAY "Some details on soil chemical analysis are
needed, as f"
@ 1,64 SAY "ollows:"
@ 3,25 SAY "CATION EXCHANGE CAPACITY"
@ 4,25 SAY "------------------------"
@ 6,9 SAY "CLASS DESCRIPTION   meq/100g
Soil"
@ 7,9 SAY "----- -----------   ---------
-----"
@ 8,11 SAY "1   Very Low
<4.5"
@ 10,11 SAY "2   Low
3.0=>5.0"
@ 12,11 SAY "3   Moderately low
5.0=>10.0"
@ 14,11 SAY "4   Medium
10.0=>15.0"
@ 16,11 SAY "5   Moderately high
15.0=>20.0"
@ 18,11 SAY "6   High
20.0=>30.0"
@ 20,11 SAY "7   Very High
>30.0"
@ 22,1 SAY "Enter Class No."
@ 23,1 SAY "Surface Horizons?"
@ 23,25 GET CECSF PICT "99"
@ 23,35 SAY "Subsurface?"
@ 23,54 GET CECSB PICT "99"
READ
CLEAR
@ 2,17 SAY "pH Measurements (in Water)"
@ 3,17 SAY "--------------------------"
@ 5,9 SAY "CLASS                pH"
@ 6,9 SAY "-----                --"
@ 8,11 SAY "1                   <4.5"
@ 10,11 SAY "2               4.5=>5.0"
@ 12,11 SAY "3               5.0=>5.5"
@ 14,11 SAY "4               5.5=>6.5"
@ 16,11 SAY "5               6.5=>7.0"
@ 18,11 SAY "6               7.0=>7.5"
@ 20,11 SAY "7                   >7.5"
@ 22,1 SAY "Surface"
@ 22,11 GET SURPH2O PICT "99"
@ 22,33 SAY "Subsurface"
@ 22,46 GET SUBPH2O PICT "99"
READ
CLEAR
@ 1,17 SAY "Available Phosphate (Bray No.2 test)"
@ 2,17 SAY "------------------------------------"
@ 3,9 SAY "CLASS RATING
RANGE(ppm)"
@ 4,11 SAY "1   Very low      <=3"
@ 6,11 SAY "2   Low           3=>6"
@ 8,11 SAY "3   Moderately low 6=>10"
@ 10,11 SAY "4   Medium        10=>15"
@ 12,11 SAY "5   Moderately high 15=>25"
@ 14,11 SAY "6   High          25=>45"
@ 16,11 SAY "7   Very High        >45"
@ 18,1 SAY "Surface horizons"
@ 18,20 GET PSF PICT "99"
READ
```

```
CLEAR
@ 1,18 SAY "Available Potassium(Ammonium acetate extract)"
@ 2,18 SAY "---------------------------------------------"
@ 4,1 SAY "CLASS              RATING
RANGE(ppm)"
@ 6,3 SAY "1                Very Low
<=30  "
@ 8,3 SAY "2                Low
30=>60"
@ 10,3 SAY "3                Medium
60=>90"
@ 12,3 SAY "4                High
90=>120"
@ 14,3 SAY "5                Very High
>120  "
@ 16,1 SAY "Surface"
@ 16,11 GET KSF PICT "99"
READ
CLEAR
@ 1,9 SAY "There is an important agroclimatic division in
Rayong a"
@ 1,64 SAY "cross a"
@ 2,1 SAY "boundary approximating to the 62 deg East Grid
line on"
@ 2,56 SAY "the latest series"
@ 3,1 SAY "L7017 topographic maps."
@ 4,9 SAY "To the East of the boundary is agroclimatic
region 2 an"
@ 4,64 SAY "d 1 lies to"
@ 5,1 SAY "the West."
@ 7,1 SAY "Region 1        At least 5 months rainy period,
suiting"
@ 7,56 SAY "most annual crops"
@ 8,16 SAY "and some fruit trees."
@ 10,1 SAY "Region 2         5-7 month rainy period suited to
most tr"
@ 10,56 SAY "ee crops."
@ 13,1 SAY "Enter the number of the region in which the land
lies?"
@ 13,58 GET AGROCLIM PICT "99"
READ
CLEAR
@ 2,9 SAY "Slope is also important in determining
suitability of t"
@ 2,64 SAY "he land"
@ 3,1 SAY "for a particular crop. To which of the slope
classes gi"
@ 3,56 SAY "ven below would you"
@ 4,1 SAY "allocate the land?"
@ 4,22 GET SLOPE PICT "!!"
@ 6,9 SAY "SLOPE %                  CODE"
@ 8,10 SAY " <2                      A "
@ 10,9 SAY "2=>5                      B"
@ 12,9 SAY "5=>8                      C"
@ 14,9 SAY "8=>16                     D"
@ 16,8 SAY "16=>35                     E"
@ 18,10 SAY ">35                     F"
READ
CLEAR
@ 3,9 SAY "That completes the information required on the
land uni"
@ 3,64 SAY "t, which"
@ 4,1 SAY "crop are you interested in (All can be chosen)?"
ACCE 'Crop Choice?' TO ASKCROP
CLEAR
@ 2,9 SAY "If you have made an error in entering the
information a"
```

```
@ 2,64 SAY "nd wish to"
@ 3,9 SAY "re-enter it, select 'A' to abandon this entry and
make"
@ 3,56 SAY "another."
@ 5,9 SAY "'Q' to abandon & quit."
@ 7,9 SAY "press any key to proceed with the evaluation."
@ 8,1 SAY ""
WAIT 'Please make your choice? ' TO VERIFY
CLEAR
IF UPPER(VERIFY)="A" .OR. UPPER(VERIFY)="Q"
DELETE
ENDIF
REPLACE DEPTH WITH LOWER(DEPTH)
ENDDO
PACK
USE LANDINF INDEX INFLU ALIAS LAND
APPEND FROM TEMP
ERASE TEMP.DBF
IF UPPER(VERIFY)="Q"
DO OPTIONS
ENDIF
SELE 5
ZAP
SELE 3
ZAP
APPEND BLANK
REPLACE CHOICE WITH ASKCROP, SELECT WITH LAND->LU
@ 20,20 SAY 'Please wait, WORKING'
RETURN
**
PROCEDURE LIBRARY
* LIBRARY.PRG
CLEAR
@ 1,27 SAY "INFORMATION"
@ 3,9 SAY "This section is provided for explanation on the
backgro"
@ 3,64 SAY "und to"
@ 4,1 SAY "the system and relevant references. Unfortunately
it ha"
@ 4,56 SAY "s not yet been"
@ 5,1 SAY "completed."
@ 7,29 SAY "SORRY"
@ 9,26 SAY "***********"
WAIT 'Please press any key to continue'
CLEAR
RETURN
**
```

APPENDIX E

Program listing
Screen dumps

```
/*      FARMTHAI.PRO  Version 1.0
        Crop selection sytem based on farmer experience
        as determined by subjective analysis of notes
        taken from interviews.

        In determining the most appropriate crop to be grown
        an assumption is made that all technical advice required will
        be available through the extension service. This is recognised
        as an over-simplification but is justified on the grounds of
        maintaining clarity in a demonstration system.

        The system uses a dynamic database, which is cleared before
        each run and updated throughout each session.

        Type RUN in response to the GOAL prompt at runtime.

COMPILER DIRECTIVES
code - increses the memory allowance for program by altering the
       size of the internal code array.

include - is used to include the contents of a separate text file
          into the program.
*/
code=3072
include "TDOMS.PRO"

/*Variable domains (types) are declared below as string, or integer,
or alist of variables, other declarations are implicit in the
declaration of predicates.*/
domains
        question,crop,factor,state = string
        value = integer
        crops=crop*
        numbers=value*
```

```
database
        posattrib(factor)
        negattrib(factor)
        logicalatt(factor,state)
        numericatt(factor,value)
        unknown(factor)
        remarks(string)

include "TPREDS.PRO"
include "MENU.PRO"

predicates
        repeatForMore
        repeatagain
        run
        add(crop,crops,crops)
        equal(crops,crops)
        select(crop)
        confirm(factor,state)
        determine(factor,value)
        condition(factor)
        affirmative(string,factor)
        negative(string,factor)
        ask(string,factor,char)
        check(factor,state)
        establish(factor,value)
        list(factor)
        record(factor,char)
        recommend(crops)
        clear_facts
        clear_unknowns
        physical(crop)
        physicalessentials(crop)
        socio(crop)
        member(crop,crops)
        member(value,numbers)
        intercrop(crop)
        legitimate(crop,crop)
```

```
run:-
        unknown(_),
        clearwindow,
        write("No information was available on the following important"),nl,
        write("queries, no decision can be made on this basis"),nl,
        noinfo,nl,
        write("Type 'Add' if you wish to add more facts and continue"),nl,
        readln(Ans),
        Ans="Add",
        clear_unknowns,
        findall(X,select(X),List),not(equal(List,[])),!,
        clearwindow,
        write("Suitable crop(s) are listed below:"),nl,
        recommend(List),nl,
        inter,nl,
        notes,nl,
        clear_facts.

run:-
        clearwindow,
        cursor(5,10),
        write("No suitable crops can be suggested for the conditions"),nl,
        write("you have specified, please check your facts. \n"),clear_facts.

repeatForMore.
repeatForMore:-
        clearwindow,
        repeatForMore.

repeatagain.
repeatagain:-
        write("not a legitimate answer, press any key"),
        readchar(_),
        clearwindow,
        repeatagain.
```

```
            recommend([]).
            recommend([Head|Tail]):-
                    write(Head),nl,recommend(Tail).

            member(Name,[Name|_]).          /*Checks for membership of a list*/
            member(Name,[_|Tail]):-
                    member(Name,Tail).

    /*The following clear_facts clauses use `retract' to remove the information
    from each database in turn, ready for a new query session. The predicate fail
    is used to force continual retraction of facts until the retract goal fails
    and so the clear_facts clause fails.  The next clear_facts clause is then resolved.
    */

            clear_facts:-
                    retract(posattrib(_)),fail.

            clear_facts:-
                    retract(negattrib(_)),fail.

            clear_facts:-
                    retract(logicalatt(_,_)),fail.

            clear_facts:-
                    retract(numericatt(_,_)),fail.

            clear_facts:-
                    retract(unknown(_)),fail.

            clear_facts:-
                    retract(remarks(_)),fail.

            clear_facts:-
                    nl,nl,write("Please press the space bar to exit"),nl,
                    readchar(_).

            clear_unknowns:-
                    retract(unknown(_)),fail.
```

```
        clear_unknowns.


        equal(X,X).


        add(Crop,[],[Crop|[]]).
        add(Crop,List,[Crop|List]).

/*The affirmative and negative clauses used below are designed to check and
update database information, posattrib stores the names of factors which have
been positively identified as present, negattrib is the converse. For updating
the ask() predicate is called.*/


        affirmative(_,Y) if posattrib(Y),!.
        affirmative(X,Y) if not(negattrib(Y)),not(unknown(Y)) and ask(X,Y,Reply),Reply='y'.

        negative(_,Y) if negattrib(Y),!.
        negative(X,Y) if not(posattrib(Y)),not(unknown(Y)) and ask(X,Y,Reply),Reply='n'.

/*The ask() predicate checks on the value of a varaible where a yes or no answer
is required and calls record() to update the relevant database, so that questions are
not repeated*/

        ask(X,Y,Reply):-
                clearwindow,
                write(X),nl,            /*Question user on factor*/
                readchar(Reply),        /*Store response to Variable*/
                write(Reply),nl,        /*confirm response on screen*/
                record(Y,Reply).        /*Update relevant database*/

        noinfo:-                        /*Writes out a list of factors for */
                unknown(Y),             /*which no information was available*/
                write(Y),nl,
                fail.

        noinfo.
```

```
inter:-
        logicalatt(intercrops,X),       /*Writes out a list of */
        write(X),nl,                    /*intercrops identified*/
        fail.

inter.

notes:-
        logicalatt(remarks,R),
        write(R),nl,
        fail.

notes.

check(slope,A):-
        repeat,
        write("What is the slope on the land considered?"),nl,
        write("Category                 Slope %"),nl,
        write("  A)                       <=2"),nl,
        write("  B)                       2-5"),nl,
        write("  C)                       5-8"),nl,
        write("  D)                       8-16"),nl,
        write("  E)                       >16"),nl,
        write("---------------------------------"),nl,
        write("Choose a category:"),nl,
        readln(Ans),write(Ans),nl,
        str_char(Ans,Choice),
        char_int(Choice,I),
        I>64,I<70,!,
        asserta(logicalatt(slope,Ans)),
        A=Ans.
```

```
check(current,C):-
        repeatagain,
        write("Which crop is currently grown?"),nl,
        write("Answer `none' if no crop is planted"),nl,
        write("----"),
        readln(Ans),
        legitimate(Ans,Crop),!,
        write(Crop),nl,
        C=Crop,
        asserta(logicalatt(current,C)).

determine(X,Y):-
        numericatt(X,Z),!,Y=Z.

determine(X,Y):-
        clearwindow,
        establish(X,Z),
        Y=Z.

previous(X,Z):-
        numericatt(X,Number),Z=Number.

previous(X,Z):-
        not(numericatt(X,_)),
        clearwindow,
        list(X),
        numericatt(X,Z).

establish(capital,C):-
        write("Does the farmer have funds to invest in a new venture?"),nl,
        write("Which category does he match as given below?"),nl,
        menu(6,10,7,7,["Substantial Savings","Creditworthy","In debt, a bad risk"],"Capital Available",
        1,Capital),C=Capital,
        asserta(numericatt(capital,C)).
```

```
establish(pineappleMarket,Dist):-
        write("What is the distance to the nearest"),nl,
        write("pineapple factory, in Km. ?"),nl,
        readint(Dist),nl,
        asserta(numericatt(pineappleMarket,Dist)).

establish(size,S):-
        write("What is the area of land under consideration?"),nl,
        write("Normally assume to be the total holding."),nl,
        write("Give to the nearest Rai---"),
        readint(S),nl,
        asserta(numericatt(size,S)).

establish(water_depth,X):-
        write("What is the depth to the water table in the dry season?"),nl,
        write("Please answer to the nearest metre>"),
        readint(X),nl,
        asserta(numericatt(water_depth,X)).

establish(loan,L):-
        write("Is a loan available from any source?"),
        menu(10,20,7,7,["Bank of Agriculture","Commercial Bank","Money lender","Pineapple Company",
        "Rubber company","Sugar company","No possibility of a loan"],"Loan Availability",7,Loan),
        L=Loan,
        asserta(numericatt(loan,L)).

establish(series,S):-
        write("Which soil series dominates the land concerned?"),nl,
        menu(10,20,7,7,["Alluvial","Chonburi","Fang Daeng","Kho Hong","Khok khian",
        "Phuket","Ruso","Ta Sae","Yala","Unknown"],"SOIL SERIES",10,Soil),
        S=Soil,
        S<>10,
        asserta(numericatt(series,S)),!.

establish(series,S):-
        S=10,
        asserta(unknown(series)).
        list(experience):-
                write("With which crops does the farmer have  previous experience?"),nl,
                write("Press return for menu choices"),
                readchar(_),
                repeatForMore,
                menu(10,20,7,7,["Coffee","Durian","Eucalyptus","Para rubber","Pineapple",
                "Vegetables","Cassava","None of these"],"EXPERIENCE (Esc to exit)",7,Crop),
                C=Crop,
                asserta(numericatt(experience,C)),
                C=0,retract(numericatt(experience,0)),!.

        intercrop("Cassava"):-
                negative("Does the land flood for more than 6 months each year? y/n or ?",long_annual_flood),
                asserta(logicalatt(intercrops,"Cassava after rice")),!.

/*Cassava rules do not need to be checked as paddy rice must already have been
selected for this rule to fire and provided the flood period in not excessive
cassava will improve annual income.*/

        intercrop("Durian intercrop"):-
                not(confirm(current,"Durian")),
                asserta(logicalatt(intercrops,"Intercrop Durian")),!.

        intercrop("Rubber intercrop"):-
                not(confirm(current,"Para rubber")),
                asserta(logicalatt(intercrops,"Intercrop Rubber")),!.

        intercrop(_).

        record(Y,'y'):-                 /*If response is yes*/
                asserta(posattrib(Y)),!./*add factor to list of +ve attributes*/

        record(Y,'n'):-                 /*If response is no*/
                asserta(negattrib(Y)),!./*add factor to list of -ve attributes*/

        record(Y,'?'):-                 /*Update listing of attributes*/
                asserta(unknown(Y)),!.  /*on which no detail can be given*/
```

```
record(_,_).                   /*No action if invalid response*/


confirm(X,Y):-
        logicalatt(X,Z),Y=Z.


confirm(X,Y):-
        not(logicalatt(X,_)),
        clearwindow,
        check(X,Z),
        Y=Z.


choose(X):-                    /*This predicate is required to check*/
        physical(X),           /*on the viability of a crop, where there*/
        socio(X).              /*is an interdependace between crops as*/
                               /*select should not be called twice*/


select("Paddy rice"):-
        physical("Paddy rice").



select("Cassava"):-            /*Choose cassava as suitable if both*/
        physical("Cassava"),   /*physical and socio-economic conditions*/
        socio("Cassava").      /*are met.*/


select("Coffee"):-
        not(condition(paddy_land)),
        physical("Coffee"),
        socio("Coffee").


select("Durian"):-
        not(condition(paddy_land)),
        physical("Durian"),!,
        socio("Durian"),
        intercrop("Durian Intercrop").
```

```
select("groundnut"):-
        not(condition(paddy_land)),
        physical("groundnut"),
        socio("groundnut").

select("Mangosteen"):-
        physical("Mangosteen"),
        socio("Mangosteen").

select("Para rubber"):-
        not(condition(paddy_land)),
        physical("Para rubber"),
        socio("Para rubber"),
        intercrop("Rubber intercrop").

select("Pineapple"):-
        not(condition(paddy_land)),
        physical("Pineapple"),
        socio("Pineapple").

select("Rambutan"):-
        not(condition(paddy_land)),
        physical("Rambutan"),
        socio("Rambutan").

condition(paddy_land):-
        posattrib(paddy_land),!.

condition(paddy_land):-
        not(posattrib(paddy_land)),
        condition(paddy_soil),
        asserta(posattrib(paddy_land)),!.
```

```
condition(paddy_land):-
        not(posattrib(paddy_land)),
        not(condition(paddy_soil)),
        condition(annual_flood),
        confirm(slope,"A"),
        asserta(posattrib(paddy_land)).

condition(paddy_soil):-
        determine(series,Soil),
        member(Soil,[1,2,5]).

condition(water_control):-
        affirmative("Are water control methods feasible, such as raised beds? (y/n)",water_control).

condition(annual_flood):-
        affirmative("Does the land flood for more than 3 months each year? (y/n)",annual_flood).

physical("Paddy rice"):-                /*Choose paddy rice as a suitable crop*/
        condition(paddy_land),          /*if paddy land & no water control*/
        not(condition(water_control)),
        intercrop("Cassava"),!.

physical("Paddy rice"):-
        condition(paddy_land),
        condition(water_control),
        not(choose("Mangosteen")),
        intercrop("Cassava").

physical("Cassava"):-
        negative("Is there evidence of soil erosion? y/n",erosion),
        confirm(current,Crop),  /*Here confirm is used to find a crop rather than check it's presence*
        Tree=Crop,              /*intermediary clause required to avoid type mismatch through database
        not(member(Tree,["Cashew","Casuarina","Coconut","Coffee","Durian","Eucalyptus","Jackfruit","La
        "Mango","Mangosteen","Para rubber","Rambutan","Sweet tamarind"])),
        confirm(slope,X),       /*Check slope is less than or equal to 5 degrees*/
        str_char(X,C),          /*as most soils in the area are susceptible to erosion*/
        char_int(C,I),          /*The member predicate above ensures there is no tree crop growing*/
        I<=66.
```

```
physical("Durian"):-
        physicalessentials("Durian"),
        determine(series,S),
        Soil=S,
        member(Soil,[3,4,6,7,8,9]).

physical("Durian"):-
        physicalessentials("Durian ),
        determine(water_depth,D),
        D<=3,
        negative("Is soil texture sandier than sandy loam?",sandy),
        confirm(slope,G),
        str_char(G,C),
        char_int(C,I),
        I<69.

physical("Para rubber"):-
        not(posattrib(paddy_land)),     /*An example of using procedural knowledge rather than relying
        determine(water_depth,D),       /*on purely declarative style. The programmer knows that questi
        D>1,                            /*on paddy land will already have been asked because of the ord
        confirm(current,C),             /*of the select(Crop) clauses.
        Crop=C,
        not(member(Crop,["Durian","Sweet tamarind","Coffee"])),
        write("Is there a risk of storm damage to"),
        write("a tall tree crop like rubber? This would"),
        write("typically occur on steep and exposed slopes."),
        negative("Yes, No or ?",stormdamage).

physical("Pineapple").

physical("Coffee"):-
        affirmative("Is there a long rainy period of 5-7 months? y/n or ?",wetclimate),
        determine(water_depth,D),
        D<=3,
        confirm(slope,S),
        str_char(S,C),
        char_int(C,I),
        I<68.


physical("groundnut"):-
        negative("Are insect pests known to be a problem for groundnuts? y/n or ?",insectpest),
        !.


physical("groundnut"):-
        affirmative("Is pesticide readily available at an affordable price? y/n or ?",pesticideUsed),
        asserta(remarks("For groundnuts pesticide is required")).


physical("Rambutan"):-
        determine(series,S),
        Soil=S,
        member(Soil,[4,7,8]),   /* Should be sandy but low lying*/
        affirmative("Is irrigation feasible? y/n or ?",irrigation).


physical("Mangosteen"):-
        determine(series,S),
        Soil=S,
        member(Soil,[2,5]).     /* Should be low lying*/



physicalessentials("Durian"):-
        confirm(current,Crop),
        Tree=Crop,
        not(member(Tree,["Cashew","Coffee","Para rubber","Sweet tamarind"])),
        negative("Is the wet season rainfall of high intensity? y/n or ?",raindrops),
        negative("Does nearby bush present a fire risk? y/n or ?",fire_risk),
        not(condition(paddy_land)).


socio("Cassava"):-
        affirmative("Is cashflow important? y/n or ?",cashflow),
        determine(capital,X),   /*Check capital is only class C*/
        X=3,
        affirmative("Is ease of management important? y/n",easy_manage),
        affirmative("Is family labour available? y/n",famlab),!.
```

```
socio("Cassava"):-
        confirm(current,"Cassava"),     /*This represents an alternative situation*/
        determine(capital,X),           /*under which cassava would be grown, because*/
        X=3,                            /*it has been the previous choice, the farm*/
        determine(size,Y),              /*is small and no risk can be taken*/
        Y<5,
        negative("Can the risk of crop failure be tolerated? y/n",risk),!.


socio("Cassava"):-
        negative("Does the farmer posess the freehold or long term tenure on his land? y/n",tenure).


socio("Durian"):-
        affirmative("Does the farmer posess the freehold or long term tenure on his land? y/n",tenure)
        determine(capital,1),
        determine(size,Y),
        Y<=50,
        negative("Is cashflow important ? y/n or ?",cashflow),
        negative("Is ease of management important?",easy_manage).


socio("Para rubber"):-
        affirmative("Does the farmer posess the freehold or long term tenure on his land? y/n or ?",te
        determine (size,Y),
        Y>50,                   /*Note how tedious the inclusion of many repeats of various      */
        determine(capital,X),   /*affirmative and negative clauses becomes in a long program     */
        X<3.                    /*Compare this to the neater confirm and determine clauses with */
                                /*their associated texts given only once.                       */
socio("Para rubber"):-
        posattrib(tenure),      /*Repetition avoided here only because this is the second part  */
        numericatt(size,S),     /*of an .OR. relationship so questions asked in the first part  */
        S<50,                   /*will have inevitably been answered.                           */
        previous(experience,Number),
        Number=4,
        affirmative("Is a regular cashflow preferred? y/n or ?",cashflow),
        determine(loan,L),
        L=5.
```

```
socio("Pineapple"):-
        determine(capital,A),
        A=1,
        determine(pineappleMarket,D),
        D<20,
        previous(experience,Number),
        Number=5,
        affirmative("Is labour available at low cost? y/n or ?",cheaplabour),  /*expensive hired labou
        negative("Is ease of management important? y/n or ?",easy_manage),        /*a constraint.*/
        affirmative("Are pineapple suckers readily available? y/n or ?",pineappleSuckers).

socio("Coffee"):-
        previous(experience,Number),
        Number=1.

socio("groundnut"):-
        confirm(current,C),
        Current=C,
        member(Current,["Para Rubber","Durian","none"]).

socio("Mangosteen"):-
        affirmative("Does the farmer have permanent tenure?",tenure).

socio("Rambutan"):-
        affirmative("Does the farmer have permanent tenure?",tenure).

legitimate(Ans,Crop):-
        member(Ans,["Paddy rice","Cassava","Durian","Para rubber","Coconut","Rambutan","Mangosteen",
        "Coffee","Sugar cane","Sweet tamarind","Pineapple","Banana","groundnuts","Mungbean",
        "Chilli","Jackfruit","Lamud","Mango","Eucalyptus","Casuarina","Cashew","Vegetables","none"]),
        !,
        Crop=Ans.

legitimate(Ans,Crop):-
        synonym(Ans,Crop).

synonym(Name,"Para rubber"):-
        member(Name,["Rubber","rubber","pararubber","para rubber","Ruber","ruber","rubbe"]),!.
```

```
synonym(Name,"Durian"):-
        member(Name,["durian","DURIAN","TOOREAN"]),!.

synonym(Name,"Pineapple"):-
        member(Name,["PINEAPPLE","pineapple","Sapparot","Saparot","sapparot","SAPPAROT"]),!.

synonym(Name,"Coffee"):-
        member(Name,["COFFEE","coffee"]),!.

synonym(Name,"groundnut"):-
        member(Name,["Groundnuts","Peanuts","GROUNDNUT"]).
```

APPENDIX F

The ITS hardware and software
Crop listing for Rayong province

Image
Technology
Systems Ltd

Aston Science Park, Aston Triangle, Birmingham B7 4BJ
Tel. 021 359 0981 Telex. 334535 BMTECH G Fax. 021 359 0433

SPECIFICATIONS ITS-30

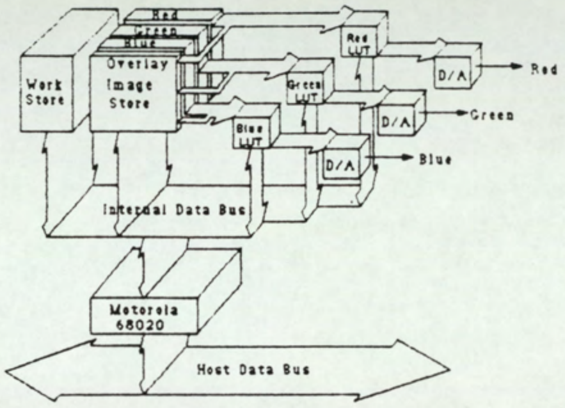
Image Processing System

BASIC HARDWARE

- 16.67MHz Intel 80386/387 based micro-computer
- 42MB 28mS Fixed Disk Drive
- 1.2MB 5.25" Floppy Disk Drive
- VGA Display & Monitor

ITS-30 SYSTEM HARDWARE

- Framestore:
  16.67MHz Motorola 68020
  1MB Workspace RAM
  768 x 768 pixel Image Memory
  3 x 8 bit image stores
  1 x 4 bit overlay store
  1-16xZoom, Pan & Scroll
- 14" Long Persistance RGB Colour Image Display
- Optical Mouse + Pad
- Colour Inkjet Printer



FRAMESTORE LAYOUT

OPTIONAL HARDWARE

- Portable host computer
- 20/25 MHz Intel 80386/387
- 300MB 16mS Fixed Disk Drive
- 20" Long Persistance RGB Colour Image Display
- Video Digitisation Module
- Colour/BW Video Camera
- Optical Storage (WORM) Device
- 1600/6250 bpi Tape Drive
- Plotters/Digitisers/Scanners
- Hardcopy Options

ITS-30 SYSTEM SOFTWARE

- ICONOCLAST Image Processing Module
- ICONOCLAST File Processing Module

OPTIONAL SOFTWARE

- ICONOCLAST Tape Support
- ICONOCLAST Video Digitisation Module
- ICONOCLAST Digital Mapping Module
- ICONOCLAST Spatial Analysis Module
- ICONOCLAST Surface Modelling Module

CROPS GROWN IN RAYONG

ASPARAGUS
BANANA
CASSAVA
CASUARINA PINE
COCONUT
DURIAN
EUCALYPTUS
GROUNDNUT
JACKFRUIT
LYCHEE
MAIZE
MANGO
MANGOSTEEN
PASSION FRUIT
PINEAPPLE
RAMBUTAN
RICE
RUBBER
SAPODILLA
SOYA BEAN
SUGAR CANE
TAMARIND
VEGETABLES
WINGED BEAN

APPENDIX G

DLD Related information
Questionnaire format
Summary of questionnaire responses
DLD land quality guidelines

13/ D(6) T(4) A(2) P(1) R(~~2~~) S(B)

64212B

Slope B

LAND EVALUATION QUESTIONNAIRE

No. 5

INTERVIEWER TRANSLATOR DATE 12/11/3.

Background

Location of farm. Muban [illegible] Sheet ______ Grid Ref. 642 2

Tambol [illegible]

Amphoe Ban Kai

Soil Series (36) Sattahip Series (Sh) Original L.U. 39 Proposed L.U. B

Typic quartzipsam

Name of farmer [illegible] Khunsom

Age(Guess & underline category)
Young(<35)/Middle aged/Senior(>60)

1.Complete the table below to indicate land tenure

Total Area Farmed (Rai)	Freehold		Permanent Ten.		Short-lease		Other	
	Area (Rai)	Charges for landuse Bt.	Area (Rai)	Rent * (Bt.)	Area (Rai)	Rent * (Bt.)	Area (Rai)	Amount (Bt.)
21	21							

*Includes Land tax?

For Other specify details ______

2. For how long has the farmer cultivated this land(years) 33 years

3. Has the farmer ever cultivated land elsewhere? If so list the locations and crops previously grown

Ban Kai - rice

4. Is the farmer relying solely on farming for his income or does he have other employment to supplement his income?

labour

5. For each crop ask why he has chosen to grow that crop? (No prompting as to possible reasons should be given by the interviewer)

No. CROP EXPLANATION

para · rubber - protect soil loss

- suggestion from official of agricultural office

cassava - around here grow cassava

sugar cane - suitable land

- near factory

- k

6. Are there any other crops the farmer has grown before, on this farm or elsewhere? If so list them and ask for explanation as to why he is not growing them now on this farm.

CROP EXPLANATION

7. For any crops mentioned on the Rayong master list but not listed under table 1 or 6/, enquire as to why the farmer does not grow that crop, list the explanations.

CROP EXPLANATION

pine apple - no market, doesn't like to grow and never

Durian
Rambutan
Mangosten } not have water pump

coffee - not understand to grow

- requirement of coffee (don't know)

co-conut - neighbour around here grow coconut doesn't [redacted] (or low yield)

8. Complete table 1 by questioning the farmer and add any additional comments here. Particularly any notes on the rotation practised or the general farming system.____________________

____

____

____

Table 1 Indicating Cropping Details

Crop	Var.	Bought seed	Area (Rai)	Sown (month)	Harvest (month)	Prior crop	Next crop	Total yld.Kg	Age tre
para-rubber	—	—	7	—	—	cassava	—	—	2.5
sugar cane	—	—	14	April	May	sugar-cane	sugar-cane	15 tons/rai 150,000 kg	—
cassava (between para-rubber)	—	—	7	April	April	cassava	cassava	2500 kg/rai	—

Table 2 only covers 10 rai

Table 2 Showing Labour Requirement for chosen Crop(hrs.)

Name of crop sugarcane (10 rai) Sale price 350 ฿/tone (0.35 ฿/kg)

		Family Labour				Exchange labour				Hired labour				Hire
ACTIVITIES	Date	No.	Days	Hrs/day	Total	No.	Days	Hrs/day	Total	No.	Days	Hrs/day	Total	Bht./day
Clearing														
Ploughing	—	—	—	—	—	—	—	—	—	—	—	—	—	240 ฿/rai
Planting	—	5	3	8	15					2	3	8	6	35 ฿/man day
Irrigation (after planting ≥ 10 days		2	—	12 hrs.	2	—		—	—	—	—			
Fertiliser aplication		5	3	8	15	—	—	—	—	—	—	—	—	—
Spray application	—	—	—	—	—	—	—	—	—	—	—	—	—	
Weeding 2 times	—	5	3	8	30	—	—	—	—	—	—	—	—	—
Harvest 2 times annual (0.05 ฿/kg)	—	1	2	8	4	—	—	—	—	—	—	—	—	50 ฿/ton
Transport	—	—	—	—	—	—	—	—	—	—	—	—	—	80 ฿/ton
Processing														
Other tasks														
Total Days (/8)					66 days									

Table 3 Showing Power Requirements Additional to Human Labour for the Specified Crop

Crop____________

ACTIVITY	Date	Draught-animal Cult.					Mechanised Cult.				
		No.Ops	Days/Op	Hrs/dy	TotHrs	Chrg.Bt.	No.Ops	Days/Op	Hrs/dy	TotHrs	Rate*
CLEARING											
PLOUGHING (		2	2	8							
TRANSPORT											
OTHER											
TOTAL(Charges)					ENTER	ENTER				?	2400 ¢

Just ENTER

9. Can the farmer give an estimate of the depth to water during the dry season for his upland? 4.5 metres

10. Which crop does he believe will have been the most profitable this year? Cassava

11. Why does he not grow more of that crop?
uses fertilizer more than sugar-cane

12. For rice identify the proportion of the yield reaching the market or elsewhere? Record the absolute amounts in appropriate units i.e. Kg

CROP	Domestic consumpt	Retained for seed	In lieu of rent	Feed/Pay labour	Market	Other

13. For each crop list where the produce is sold & when.

CROP	MARKET	Month of Sale	Transport Cost (Bht/Kg)	
sugar cane	- Ban Kai factory (28 km)		50 ฿ / 1000 kg	
	- Sriracha		90 ฿ / 1000 kg	for sometim
	Chonburi		100 —	
cassava	Ban Kai		10 ฿ / 100 kg	

14. For any of the crops grown are there periods of labour shortage? If so list below.

CROP	ACTIVITY	MONTH	EXTRA LABOUR REQ.
sugar-cane	harvest	May - June	90 ฿ / 1000 kg
			(50 ฿ / 1000 kg — norma

15. List all animals owned by the farmer and their use.

2 cows — hallowing { for - sugar can
cassava

16. Does the farmer have to borrow money to farm? If so from whom? capitalist

17. For how long must he normally take the loan and at what rate of interest? 3 % / month one year

Summary of Questionnaire Responses

MOIST	SERIES	SLOPE	AGE	TEN	AREA	EXP	OTHEXP	EMP	CAP	FAMLAB	HIRATE	OXEN	TRACT	TILLER	WATER	PUMP	LOAN	PRIV	BANK	CROP
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	CASSAVA
F	PIEDMONT	B	2	P	50	4	Klaeng	F	B	F	50	T	T	F	8	F	T	F	T	CASSAVA
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	CASSAVA
F	UNDULATING	B	2	F	20	6	Chonburi	F	C	T	40	T	T	F	3	F	F	F	F	CASSAVA
F	EROSION	ABD	1	F	29	20	Paddy rice	F	B	T	30	T	F	F	6	T	F	F	F	CASSAVA
F	UNDULATING	AB	1	F	9	5	Cassava	F	B	T	40	T	T	F	3	F	F	F	F	CASSAVA
F	ALLUVIAL	A	2	F	2	24	no	F	C	T	40	F	F	F	2	F	F	F	F	CASSAVA
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	CASSAVA
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	COCONUT
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	COCONUT
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	COCONUT
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	COCONUT
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	COCONUT
F	UNDULATING	A	3	F	38	50	Pado, rice	F	B	T	40	T	F	T	2	T	F	F	F	COCONUT
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	COCONUT
T	PIEDMONT	B	2	F	48	20	no	T	A	F	46	F	F	F	2	T	F	F	F	COFFEE
T	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	COFFEE
T	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	COFFEE
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	DURIAN
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	DURIAN
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	DURIAN
F	EROSION	BDC	2	F	30	7	Rubber	T	A	T	40	F	T	F	5	T	F	F	F	DURIAN
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	DURIAN
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	DURIAN
F	PIEDMONT	B	2	F	48	20	no	T	A	F	46	F	F	F	2	T	F	F	F	MANGO
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	MANGO
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	MANGO
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	MANGOSTE
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	MANGOSTE
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	MANGOSTE
F	UNDULATING	B	3	F	20	35	Paddy rice	F	B	T	40	T	T	F	4	T	T	F	T	PADDY RI
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	PADDY RI
F	ALLUVIAL	A	1	F	14	35	no	F	B	F	40	F	F	T	3	F	T	F	T	PADDY RI
F	ALLUVIAL	A	3	F	4	61	no	F	B	T	40	T	T	F	2	T	F	F	F	PADDY RI
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	PADDY RI
F	UNDULATING	A	3	F	38	50	Paddy rice	F	B	T	40	T	F	T	2	T	F	F	F	PADDY RI
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	PADDY RI
F	EROSALLUV	AB	2	F	19	18	Paddy rice	T	A	F	40	F	F	F	2	F	T	F	T	RAMBUTAN
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	RAMBUTAN
F	PIEDMONT	AB	2	F	50	3	Rubber	F	A	T	40	F	F	F	6	T	T	F	F	RAMBUTAN
F	EROSION	B	3	F	10	8	Durian	F	A	T	40	F	F	F	2	T	T	F	T	RAMBUTAN
F	EROSION	B	2	F	14	20	no	F	B	F	77	T	T	F	6	F	T	F	T	RUBBER
F	EROSION	C	1	P	41	21	Rubber	F	A	T	44	F	F	F	9	F	F	F	F	RUBBER
F	PIEDMONT	B	2	P	50	4	Klaeng	F	B	F	50	T	T	F	8	F	T	F	T	RUBBER
F	ALLUVIAL	AB	2	F	50	25	Rubber	F	A	F	56	F	F	F	3	T	F	F	F	RUBBER
F	PIEDMONT	B	2	F	50	15	Rubber	F	A	F	66	F	F	F	4	F	F	F	F	RUBBER
F	EROSION	B	2	F	100	20	Cassava	F	A	F	48	F	F	F	7	F	T	F	T	RUBBER
F	EROSION	B	2	F	41	10	Cassava	F	B	F	48	T	T	F	2	F	T	F	T	RUBBER
F	UNDULATING	B	2	F	21	32	Paddy rice	F	B	T	35	T	T	F	4	T	T	T	F	RUBBER
F	ALLUVIAL	A	1	F	14	35	no	F	B	F	40	F	F	T	3	F	T	F	T	RUBBER
F	EROSION	ABD	1	F	29	20	Paddy rice	F	B	T	30	T	F	F	6	T	F	F	F	RUBBER
F	EROSION	AB	2	F	10	30	no	F	B	F	25	F	F	F	6	F	T	F	T	RUBBER
F	UNDULATING	A	3	F	38	50	Paddy rice	F	B	T	40	T	F	T	2	T	F	F	F	RUBBER
F	EROSION	BDC	2	F	30	7	Rubber	T	A	T	40	F	T	F	5	T	F	F	F	RUBBER
F	EROSION	B	2	F	90	18	no	T	B	F	30	T	T	F	2	F	T	F	T	RUBBER

Anpory.

How to identifity land units.

แนวทางการกำหนดหน่วยที่ดิน

Mr kamron Trifak

คำรณ ไทรฟัก DLD

JUNE

มิถุนายน 2528 Land Use planning & P..

Scu I

การวางแผนการใช้ที่ดินจำเป็นต้องใช้ข้อมูลดินจากแผนที่ดิน แต่หน่วยแผนที่ดิน ได้แก่ ชุดดินต่าง ๆ มีเป็นจำนวนมากที่มีศักยภาพในการใช้ประโยชน์ที่ดินเหมือนกัน ขณะเดียวกันบางชุดดินก็มีศักยภาพต่างกัน แม้ว่าจะเป็นหน่วยแผนที่ดินเดียวกัน ดังนั้นจึงต้องมีการกำหนดหน่วยที่ดิน (Land Units) โดยคำนึงถึงคุณสมบัติทางกายภาพ ได้แก่ ความลาดชันของพื้นที่ การระบายน้ำ เนื้อดินบนและดินล่าง มาพิจารณาร่วมกันดังนี้

ตารางที่ 1 Drainage class × slope class

	% slopes	0-2	2-5	5-8	8-16	16-35	>35
Drainage class		A	B	C	D	E	F
Very poorly, poorly	1	1 A	1 B	1 C	1 D	1 E	1 F
Somewhat poorly	2	2 A	2 B	2 C	2 D	2 E	2 F
Moderately well	3	3 A	3 B	3 C	3 D	3 E	3 F
Well	4	4 A	4 B	4 C	4 D	4 E	4 F
Excessively	5	5 A	5 B	5 C	5 D	5 E	5 F

ปัจจัยที่สามที่นำมาพิจารณา คือ เนื้อดิน เนื่องจากการใช้ประโยชน์ที่ดินสำหรับการปลูกพืช เนื้อดินจะมีอิทธิพลทั้งโดยตรงและทางอ้อมต่อการเจริญเติบโตของพืช เพราะเป็นตัวควบคุมการหยั่งของราก ความสามารถในการอุ้มน้ำ การซาบซึมน้ำ เป็นต้น ในการพิจารณากำหนดหน่วยที่ดิน จะจำแนกเนื้อดินออก 2 ระดับความลึก คือ 0-30 ซม. และ 30-100 ซม. เพราะจะมีผลต่อพืชรากลึกและรากตื้น โดยจำแนกเนื้อดินตามกลุ่มของเนื้อดิน (Particlc size classes) กล่าวคือ พวก sandy, coarse-loamy, fine-loamy, coarse-silty, fine-silty, clayey, และ skeletal และ matching เนื้อดินบนและดินล่างดังตารางที่ 2

ตารางที่ 2 เนื้อดินบน (0-30 ซม.)/เนื้อดินล่าง (30-100 ซม.)

เนื้อดินล่าง เนื้อดินบน		Sandy 1	Cos. loamy 2	fine loamy 3	cos. silty 4	fine silty 5	clayey 6	skeletal 7
Sandy	1	11	12	13	14	15	16	17
Cos. loamy	2	21	22	23	24	25	26	27
fine loamy	3	31	32	33	34	35	36	37
Cos. silty	4	41	42	43	44	45	46	47
fine silty	5	51	52	53	54	55	56	57
Clayey	6	61	62	63	64	65	66	67
Skeletal	7	71	72	73	74	75	76	77

More 35% vol gravel

ตัวอย่างการจำแนก LU จากแผนที่ดิน Nn-e1A

เพราะว่าดินชุดนครพนมมีการระบายน้ำค่อนข้างเลว ตารางที่ 1 จึงจัดอยู่ในพวก 2A และดิน Nn เป็นพวก clayey คือ ดินล่าง (B horizons) เป็นพวก clayey จากตารางที่ 2 จึงเป็นพวก 36 ดังนั้น ดิน Nn-e1A จึง code ได้เป็น 2A36 ซึ่งจากตารางที่ 1 และ 2 สามารถแปลความหมายได้ดังนี้ :-

2 = มีการระบายน้ำค่อนข้างเลว

A = ความลาดชัน 0-2%

3 = เนื้อดินบนเป็นพวก fine-loamy

6 = เนื้อดินล่างเป็นพวก clayey

จาก Code 2A36 นี้จะกำหนดได้เป็นหนึ่ง LU ดังนั้นมีดินหลายชุดดินเมื่อให้ Code แล้ว จะเป็น LU เดียวกัน ขณะเดียวกันดินชุดเดียวอาจจะแยกคนละ LU ได้ เช่น ดินที่กำหนดความลาดชันไว้กว้างกว่าชั้นของ slope นอกจากลักษณะสำคัญ 3 ประการที่นำมากำหนด LU ดังกล่าวแล้ว ปัจจัยอื่นที่มีผลต่อศักยภาพในการใช้ที่ดิน เช่น ความเสียหายจากน้ำท่วม ความเค็ม การมี jarosite ความล์าะของดิน เป็นต้น เช่น 1A66 แต่มี jarosite ที่ 70 ซม. ก็จะเป็น $\frac{1A66}{j\ d_3}$ c ซึ่งหมายถึง

1 = มีการระบายน้ำเลวถึงเลวมาก

A = ความลาดชัน 0-2%

6 = ดินบนเป็นพวกดินเหนียว

6 = ดินล่างเป็นพวกดินเหนียว

j = มีชั้น jarosite

d_3 = ความลึก 50-100 ซม. ?

ในกรณีของดินตื้น ยกตัวอย่างเช่น Kt-$\frac{slB}{d_3}$ หรือ Code 3B23d_3

3 = การระบายน้ำค่อนข้างดี

B = ความลาดชัน 2-5%

2 = ดินบนเป็นพวก coarse ~~fine~~-loamy

3 = ดินล่างเป็นพวก fine-loamy

d_3 = ความลึก 50-100 ซม.

ในกรณีที่ต้องการบอกให้ทราบว่า ชั้นที่พิกัดความลึกของดิน นั้น เป็น skeletal ชั้น Lithic หรือ Paralithic ก็สามารถทำได้โดยเพิ่ม Code เข้าไป ถ้าพิจารณาแล้วเห็นว่าจะทำให้ศักยภาพในการใช้ประโยชน์ที่ดินไม่เหมือนกัน เพื่อจะได้กำหนดเป็น LU ใหม่ เช่น 3B23d_3s 3B23d_3l 3B23d_3p เป็นต้น

นอกจากนี้ศักยภาพของดินยังขึ้นกับคุณสมบัติของดินอื่น ๆ ที่มิได้นำมาประเมินในการกำหนด LU เนื่องจากซับซ้อนเกินไป แต่ถ้ามีความจำเป็นจริง ๆ ก็สามารถ code เพิ่มได้ เช่น เดียวกับ ความเค็ม jarosite ความลึก ดังกล่าวแล้ว ยกตัวอย่างเช่น ดิน 1A66 จากตารางที่ 2

จะทราบว่าดินนี้เป็นดินเหนียวทั้งดินบนและดินล่าง โดยความจริงแล้ว 1A66 อาจประกอบด้วยชุดดินที่มีชนิดของอนุภาคดินเหนียวแตกต่างกัน ก็อาจเป็น kaolinite หรือ mont-morillonite เมื่อคำนึงถึงด้านการจัดการที่ดินทั้งสองชนิดนี้อาจมีศักยภาพแตกต่างกันเด่นชัด กรณีเช่นนี้ควรแยกคนละ LU คือ 1A66k และ 1A66m เป็นต้น

อย่างไรก็ตามในการกำหนด LU ควรจะยึดตารางที่ 1 และ 2 เป็นหลัก จากนั้นค่อยพิจารณาในแต่ละ LU ว่าประกอบด้วยชุดดินอะไรบ้าง มีคุณสมบัติเหมือนกันหรือแตกต่างกันเพียงใด เช่น ความอุดมสมบูรณ์ของดิน ถ้าคุณสมบัติที่แตกต่างนั้นมีผลต่อศักยภาพการใช้ประโยชน์ที่ดินก็ควรจะแยกเป็น LU ใหม่ ดังวิธีการที่กล่าวมาแล้ว นอกจากนี้มีคุณสมบัติอื่น ๆ อีกหลายอย่างที่มิได้นำมากล่าว ณ ที่นี้ เนื่องจากจะทำให้ LU ละเอียดปลีกย่อยมากเกินไป ไม่เหมาะกับการวางแผนระดับจังหวัด แต่จากแนวทางในการกำหนด LU นี้ เมื่อนำไปเทียบกับพิกัดของความต้องการของพืชเฉพาะอย่างในการประเมินระดับความเหมาะสมของ LU ต่อพืชชนิดนั้น ๆ หรือ LUT (Land Utilization Type) แล้วจะทราบว่า LU ที่กำหนดขึ้นควรจะมีการเพิ่มหรือลดเพียงใด

<u>เอกสารอ้างอิง</u>

1. "Soil Taxonomy", 1975. Agricultural Handbook No.436

2. "Guidelines : Land Evaluation for Rainfed Agriculture"
FAO Soil Bulletin 52, FAO Rome, 1983.

3. มาตรฐานการสำรวจดินค่อนข้างละเอียด
เอกสารวิชาการ กองสำรวจดิน. 2522

Appendix

1. Slope Class

%	class	Complex slope
0-2	A	nearly level to level
2-5	B	undulating
5-8	C	gently rolling
8-16	D	rolling
16-35	E	hilly
35-50	F	steep
50-75	G	very steep
> 75	H	extremely steep

5C ?

2. Drainage class

Drainage class	Code	Subgroups	Major land type
Very poorly or poorly drained	1	Typic of Aquic ~~sub~~ suborders	Paddy
Somewhat poorly drained	2	Aeric of Aquic suborders	Paddy
Moderately well drained	3	Typic and/or Aquic of Ustic or Udic suborders	Upland
Well drained	4	Typic of Ustic or Udic suborders	Upland
Excessively well drained	5	Typic of Psamment suborder	Upland

3. Depth phase

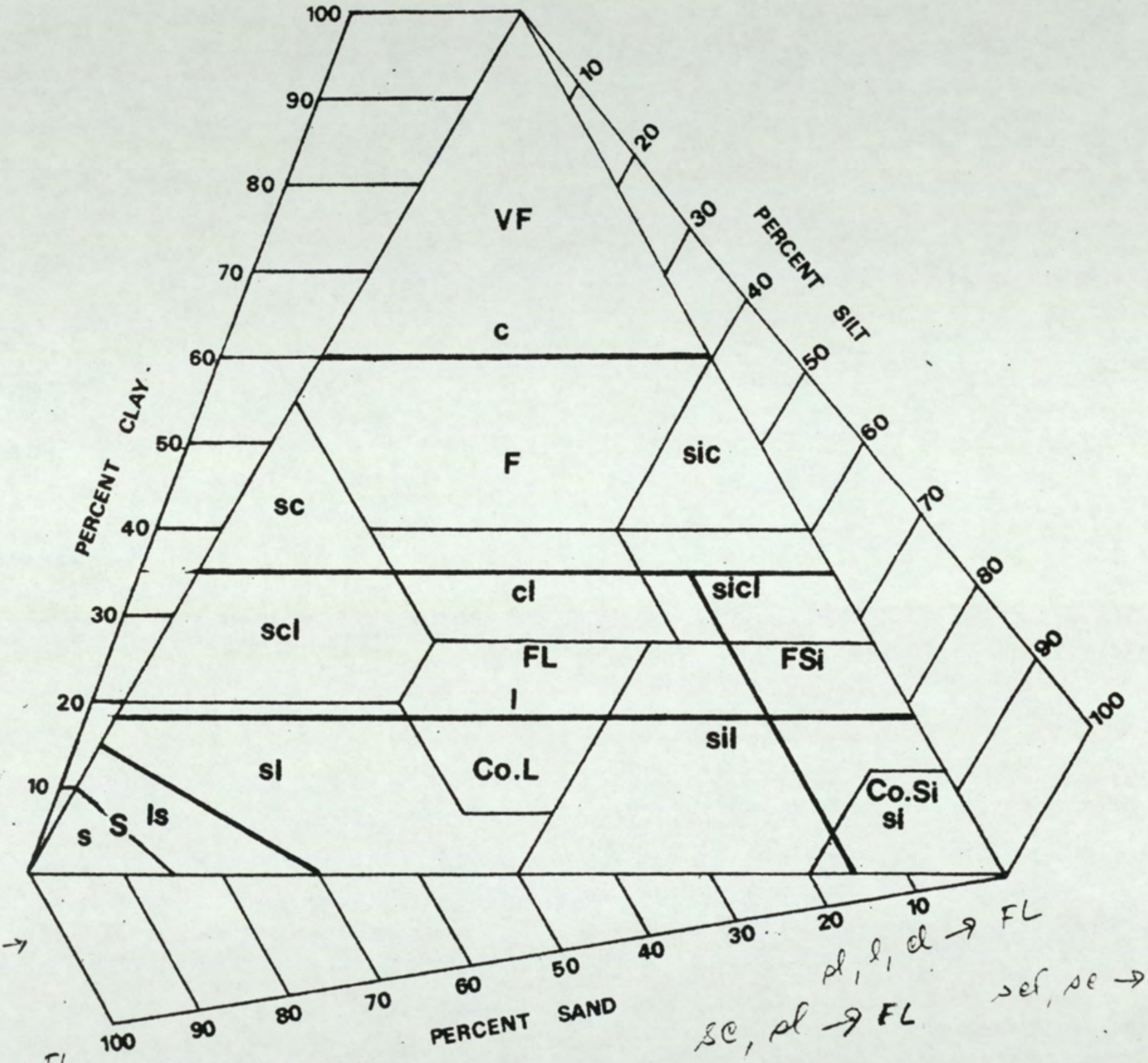
cm.	class	Description
0-15	d_1	very shallow
15-50	d_2	shallow
50-100	d_3	moderately deep
100-150	d_4	deep
> 150	d_5	very deep

4. Particle size class

Particle size class	clay %	0.25-2 mm. Ø %	Texture	Remark
Sandy	< 18	-	sand, loamy sand	
coarse-loamy	< 18	> 15	sandy loam, loam[1] silt loam	
fine-loamy	18-34	> 15	sandy clay loam, loam[1], clay loam[2], silty clay loam[3]	
coarse-silty	< 18	> 15	silt, silt loam[4]	
fine-silty	18-34	> 15	silt loam[4] silty clay loam[3]	
clayey : fine	* 35-59		sandy clay, clay loam[2] silty clay loam[4], silty clay, clay[5]	* 30-59 in Vertisols
very fine	≥ 60		clay[5]	

1,2,3,4,5 ดูเปรียบเทียบในตารางสามเหลี่ยม ในการกำหนด Particle size classes

sl, scl → FL
scl, sc → FL



cl, sicl, c →

sl, scl → FL

ls, sl → CoL
scl, sc, c → F
cl, c → F

sc, c → F

l, cl → FL
cl, l, sicl → FL

sl, l, cl → FL
sc, sl → FL
scl, sc → FC

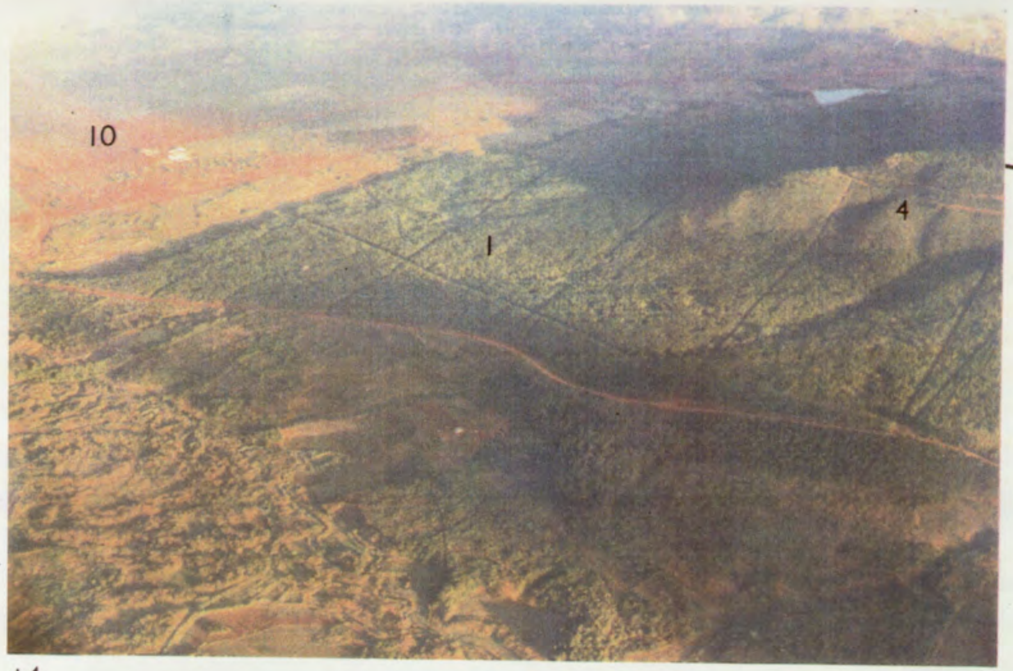
ไดอะแกรมสามเหลี่ยมมาตรฐาน

sl, l → CoL
sl → FL

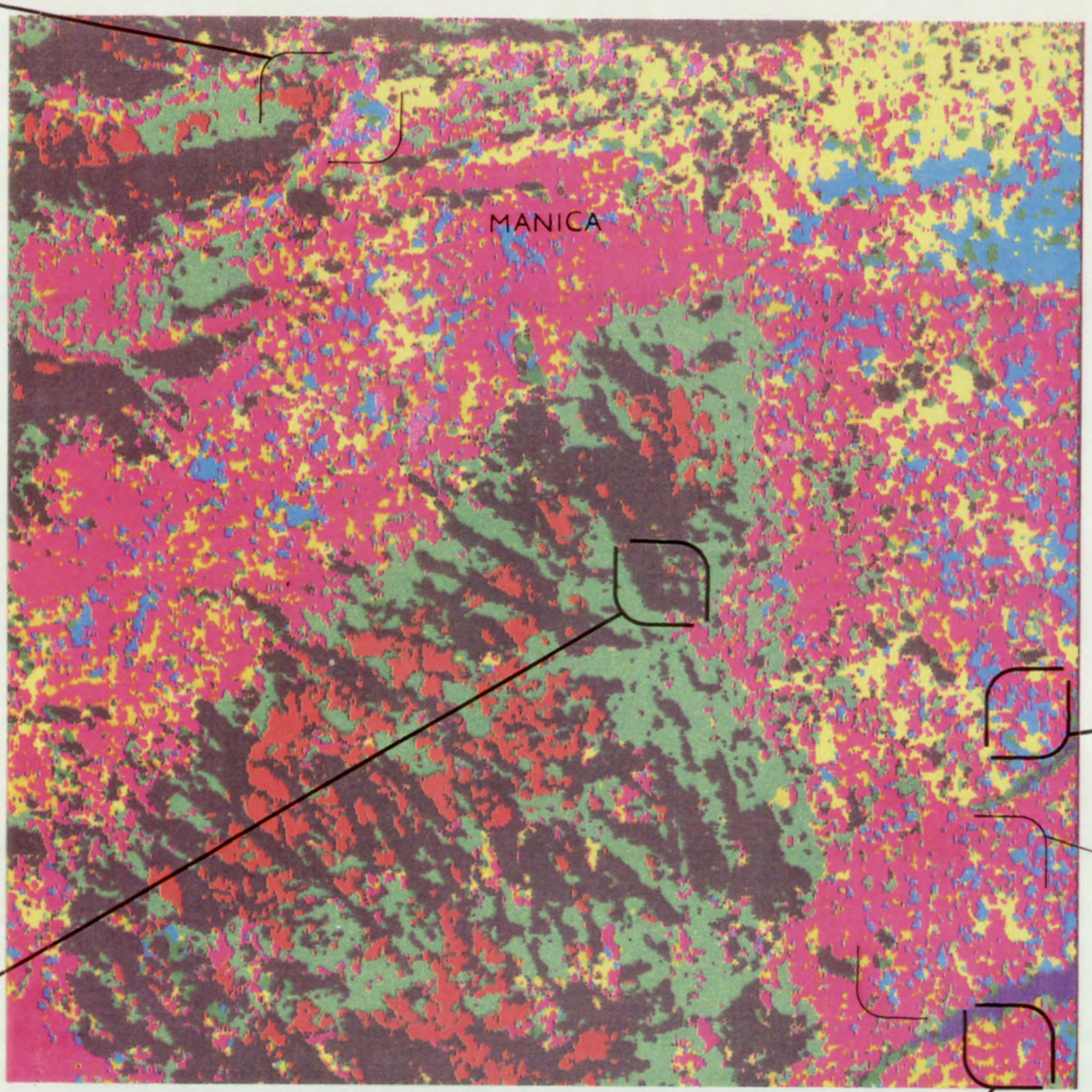
EXAMPLE SET OF PLATES ILLUSTRATING MOZAMBIQUE LANDSAT MSS
CLASSIFICATION APPLIED TO MANICA PROVINCE
(Plates 5.14 - 5.18 related to section 5.7.2)

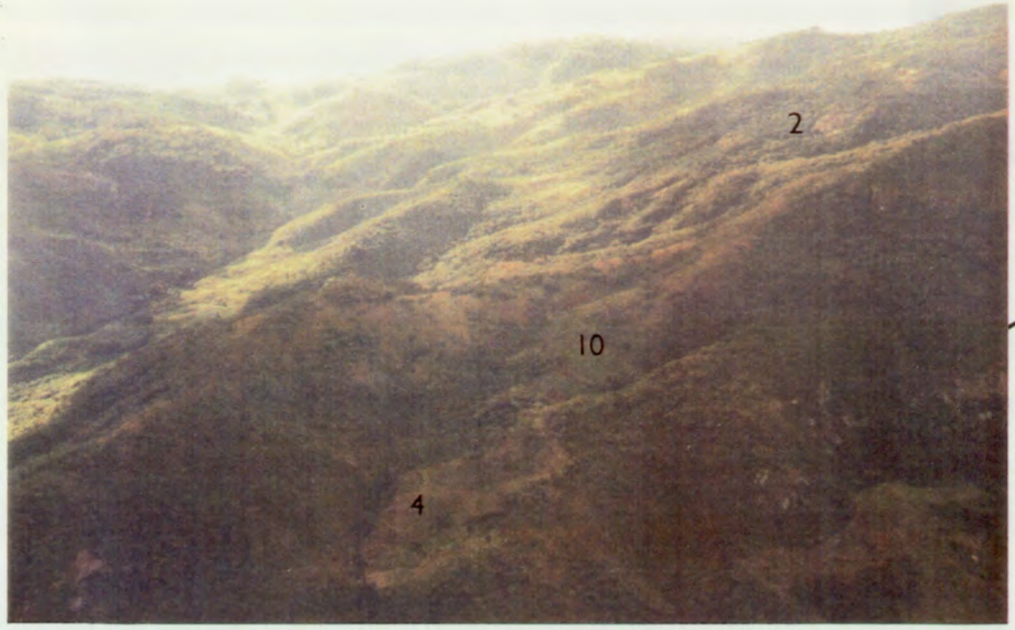
The University of Aston in Birmingham
COMPUTER AUTOMATION OF LAND EVALUATION PROCEDURES
M R Pooley
Doctor of Philosophy
1989

EXAMPLES

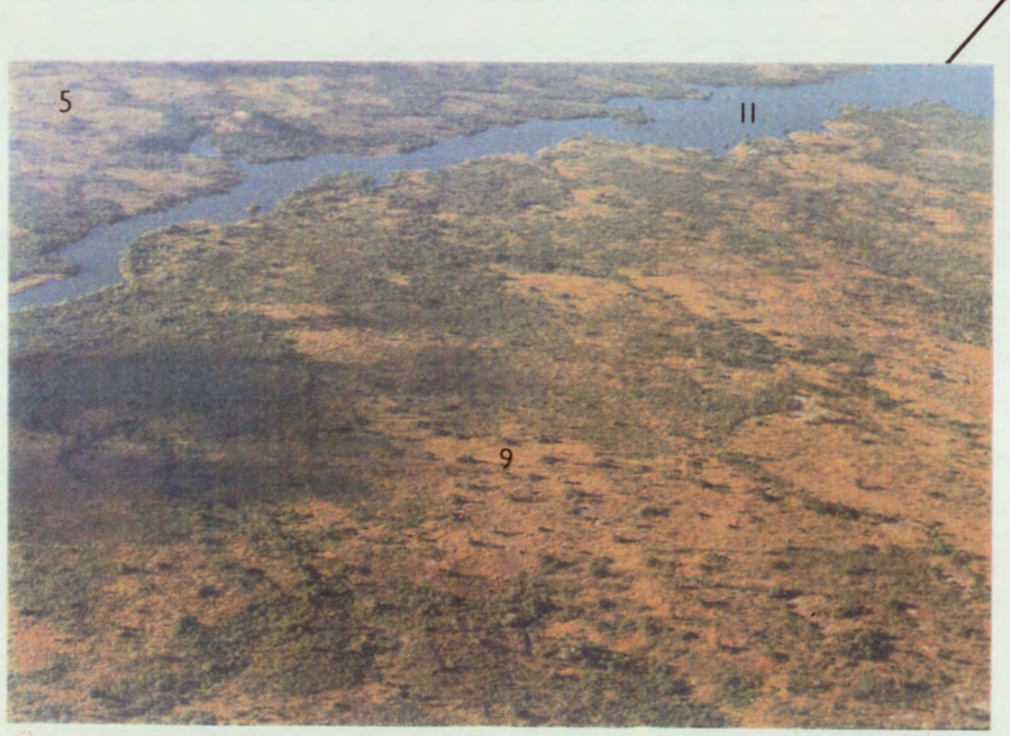


.14



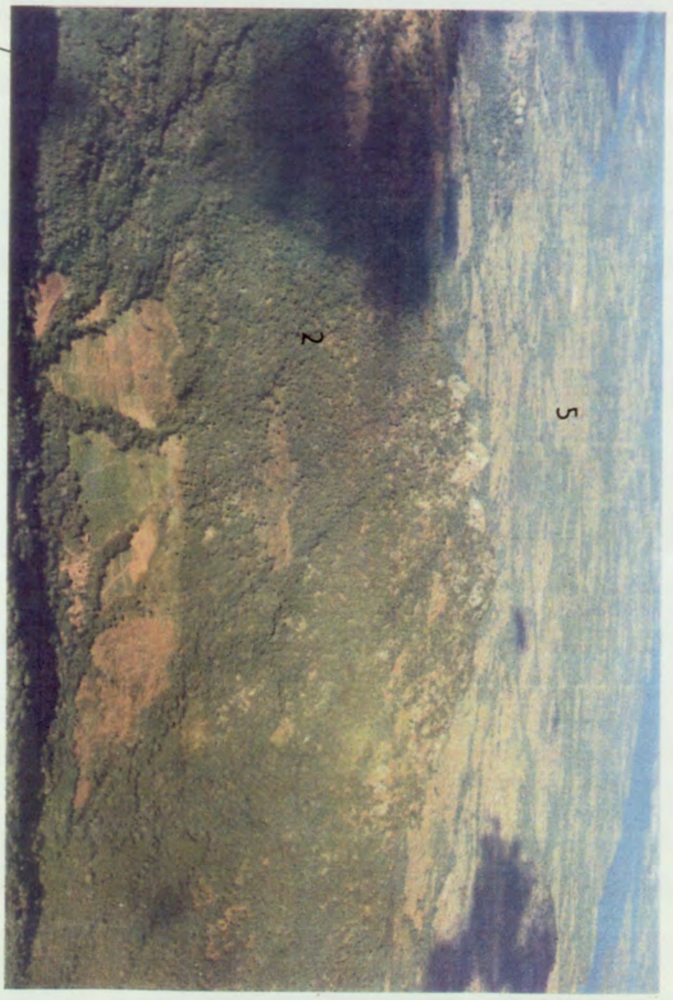


5.15



5.16

5.18



5.17