SYSTEMS IN FISHERIES MANAGEMENT AND RESEARCH

FISHERIES MANAGEMENT EXPERT SYSTEMS

THEORETICAL CONSIDERATION AND PRODUCTION

OF PROTUNA AND PROFISH

PRODUCED BY MRAG LTD
UNDER THE
FISH MANAGEMENT SCIENCE PROGRAMME
OF THE
OVERSEAS DEVELOPMENT ADMINISTRATION

DECEMBER 1991

CONTENTS

1 INTRODUCTION

OBJECTIVES

Originally this project for the development of a Fisheries Management Expert System (FMES) was proposed as a solution to two interrelated problems that often occur when developing a coherent fisheries management strategy.

(i) Available management options and their information requirements are not well known by those charged with policy development and implementation.

(ii) Extensive data collection programmes may have been in place for several years, but the use of this data in appropriate analyses that can provide information to managers is difficult and rarely undertaken effectively.

Although highly trained people and microcomputers may often be available, there are few analytical tools to guide and assist stock assessment and fishery management personnel. In many cases, a clear link between fish stock assessment and fisheries management policy is lacking, even when there is good data and well defined management objectives.

The FMES aims to provide computer based tools for fish stock assessment and fisheries management policy. Stock assessment and analyses which are particularly appropriate for the stocks of the target countries are linked to management policy decisions through a fisheries management "expert system" computer program.

Assessment methods for tropical fisheries must usually be based on the size, rather than the age, composition of a stock because spawning and feeding are continuous as opposed to the cyclical patterns of temperate species that arise from seasonality.

While several new assessment methods have been developed along these lines, the computer software is not available in a user-friendly form and the output from these size composition based methods are not directly linked to management options. This creates a significant barrier between the fishery scientist analyzing the data, and the policy and decision makers seeking advice.

Expert systems software can alleviate this problem by providing a guiding framework for both scientists and managers to step through assessment and management options with better decision making capabilities.

APPROACH

A two pronged approach has been developed with :

(i) Expert systems being developed as an aid for management decision making, including advice on suitable methods of assessment.

(ii) Fish stock assessment software using length frequency distribution analysis (LFDA) and production modelling.

This report describes the development of the expert systems FMES ProTuna v2 and FMES ProFish v3, both are fully documented with accompanying user manuals.

The final report produced here is for this expert systems component of the project. The stock assessment component of the work is presented as a separate report and product.

The two expert systems products annexed to this report are related as follows :

FMES ProTuna v2

This is specialized in the narrower domain of fisheries management for tuna fisheries and is implemented as a simple two stage process :

(i) The consultation which draws up a list of recommended management plans of action.

(ii) The running of these plans to deliver appropriate and relevant advice.

FMES ProFish v3

This is a more generalised fisheries management expert system not specializing in any one species and provides a broader range of advice concerning the most likely resource species in the target area. ProFish v3 builds on the experience gained in developing the ProTuna system and feedback from potential users. As a result it employs a more flexible design in keeping with the working methods of fisheries managers. It is broken up into more stages allowing the user a natural break to complete external activities recommended at each stage. The results of these recommended activities can then be used as input into the next stage which can be run later at the users convenience. ProFish asks for and makes use of more detailed information on fishery characteristics, policy environment, current management measures and management objectives.

THEORETICAL CONSIDERATIONS

Though expert systems are to many an attractive notion they are often poorly understood; that is in themselves, apart from the actual advice they provide. A clear explanation of their origins, capabilities and limitations is useful to put the present project into context and give clearer ideas for any future work. A thorough review of theoretical considerations is given in Annex 1.

2 CHOICE OF DEVELOPMENT SOFTWARE

Both expert systems have been designed to run on a standard Personal Computer (P.C.) i.e. 286/386 running at 6-20 MHz, with 512 K or preferably 640K RAM, a hard disk and a variety of graphics controllers and standard peripherals i.e. printer and may be used with or without a mouse. These are the typical types of platform that are available in fishery departments for managers to use. A similar hardware environment was available for the development work and this placed some constraints on the software that could be used in implementing the design of the systems given the developers limited experience of the different types of software available and the time available to develop the prototypes.

These constraints ruled out the use of workstation or mainframe development environments and the use of toolkits. Non-procedural languages both functional (LISP) and logic based (PROLOG) and object oriented languages used for implementing structural objects (e.g. LOOPS) were also ruled out. A conventional procedural language (e.g. Pascal) could have been used as an alternative, however, a considerable time penalty would have been incurred in creating a robust inference engine with a reliable conflict resolution strategy.

For these reasons a decision was made to use a commercial shell with an inbuilt inference engine and added facilities for rapidly developing an effective user interface. Commercial shells are designed to run well on P.C.'s. Characteristically, they can impose a rigid knowledge representation and inference strategy on the development e.g. most will only use a rule based method of language representation with factual knowledge simply coded in as constants. More often than not they will also only have a standard conflict resolution strategy. This feature may be useful to the developer initially in limiting potentially confusing options but may hinder flexibility at a later stage when the developer begins to understand the potential use of such options but cannot gain access to them.

Taking all these considerations into account "KnowledgePro" was chosen as the preferred medium of development. It is not strictly a shell but more of a high level development language aimed at the construction of expert systems. It is not as restrictive as some software e.g."Crystal" and offers some advantages which aid the rapid development of an application. It has most of the constructs and facilities of a high level language such as PASCAL but offers additional list processing, data validation, and interface facilities. It also allows limited object representation, "topics" which are the basic programming unit can be procedures as well as variables and even both at the same time. KnowledgePro comes equipped with comprehensive 'Hypertext' support. KnowledgePro provides the developer with a turbo pascal type working environment with menu control over a compiler, debugger and wordstar like code editor. Code is not automatically generated (apart from with some of the graphics tools) but is written and compiled, this feature provides additional flexibility. The default conflict resolution strategy is backward chaining both upwards and outwards, i.e. to parent topics and siblings of parents. This can be overridden by explicitly stating search paths.

Though limited object representation is allowed via topics, it is not possible to construct structural objects using rule based representations. The assumed method of knowledge representation and inference is the production rule. As already mentioned a rule based system is an instinctive first choice when trying to grasp the problems of knowledge representation, though in the long term it may not be the best or only form that should be used it allows the developer to tackle a problem reasonably quickly.

HYPERTEXT

Hypertext can be described as the opposite of an expert system. The latter asks for information and makes a decision, whereas hypertext provides information for the user to make decisions. A combination of the two technologies can greatly enhance the usefulness of an application.

Using hypertext any screen information presented to a user can have key words or groups of words back highlighted. If all of these are understood then the user can progress as normal. However, if the user needs further explanation of any of these key expressions, it can be selected with a mouse or function key and 'activated'. This procedure will bring up a new screen explaining the offending term or action required by it. This screen in turn can have yet another level of explanation for any of its words that are ambiguous. These textural links can be nested to any number of levels and horizontally cross referenced to each other and simple or complex relationships can be built up between blocks of information.

In KnowledgePro these blocks are classed as 'topics', the same as a functional programming unit. It enables control of the programme to also branch through hypertext to link into a smaller 'sub expert system' . This in turn can provide further dynamic advice, rather than the user being left to view an unchanging screen of textural explanation. Hypertext is used as an explanation facility in ProTuna to provide information on terminology as well as causal knowledge.

In a field as complex as fisheries management, such a combination of technologies can help ensure that :

a) The right kind of questions are asked; and

b) The user gives a sensible response.

3 PRODUCT DEVELOPMENT

INTRODUCTION

Given a scheduled time horizon for product development of 2½ years from May 1989, it was decided that the best method of product development for this novel form of expert system was the production of a series of prototypes. The first full working version of ProTuna, aimed to provide fisheries managers with advice and guidance in the management of their fisheries that integrated the correct application of fisheries science into the overall management process. This advice when related to fisheries science concentrated on areas relevant to a set of accompanying assessment software for length based analytical models and production modelling. ProTuna was designed for use in the island states of the Pacific and Indian Oceans and concentrated on Tuna management, these being the most important commercial species. In the process of the development of ProTuna there was a background objective that the design be general enough for use in other spheres of fisheries management i.e. other species and areas. Eventually it was decided that these latter aims could best be achieved successfully with a redesigned package which is embodied in the product ProFish.

SUITABILITY OF FISHERIES MANAGEMENT FOR EMBODIMENT IN AN EXPERT SYSTEM

The construction of an effective expert system in any domain is an extremely complex task and entails much more than coordinating the use of existing software. Yazdani (in Forsyth 1986) lists the following as difficult properties of knowledge and experience when assessing the suitability of a domain for expert systems :

1 Experts generally do not agree with one another at all.

2 Strategies in reasoning are very complicated.

3 The knowledge includes temporal relationships.

4 Problems take a long time to be solved by people.

5 A lot of actions hinge on a lot of objects.

6 Too many objects and too much reliance on common concepts.

This is a somewhat depressing catalogue of unsuitable criteria especially when considering the typical types of problems that can occur in fisheries management. However many of these problems are probably not insurmountable. The very process of rigorous analysis of the domain will reveal inconsistencies and contradictions and resolution of these will aid progress towards more rational management.

A practical outcome of the development of these expert systems is that attention has been focused on the whole process of fisheries management and on how best to design a logical and coherent representation of it that is both implementable and usable.

CHOICE OF EXPERT

The choice of expert is critical in the development of an application. It is essential to have a expert who is both enthusiastic and cooperative. Dr J.A.Gulland FRS provided the majority of management advice for the initial prototypes. Sadly, Dr Gulland died just after the first phase of the project was completed and further expertise was enlisted for the continuation of the project. Expert knowledge was elicited from David Evans, Andy Rosenberg, John Beddington and others. Cradoc Jones formalised this knowledge into the framework and rule systems of ProTuna and ProFish.

METHOD OF KNOWLEDGE ACQUISITION

The informal ad-hoc method outlined by Yazdani (Forsyth 1986) for knowledge acquisition from the experts was used. This method is a somewhat trial and error approach but has previously led to the development of successful applications and is useful when first experimenting with the development of expert systems. The basic methodology is :

(i) Structure the domain by forming a good model of the experts problem solving processes.

(ii) Produce a rough and ready prototype which will reveal problems and the refinements that are required.

(iii) Loop around the refining/testing/debugging cycle as many times as necessary until the system is satisfactory or until you have learned enough to design a new version.

STRUCTURING THE DOMAIN

It is difficult to get experts, particularly when they are first confronted with knowledge engineering techniques, to come up with ideas from cold. It is a daunting prospect for an expert to rationalise his hard earned experience into the concise format of a computer programme. Most experts have a 'feel 'for their domain and it is difficult to rationalise this and avoid loose ends and contradictions. Experts, particularly scientists, often try to make their expertise respectable by trying to express it within some artificial framework, i.e. the scientific method. This is more than likely not a true representation of their actual methods.

One method of making expert more comfortable with knowledge engineering is to present the expert with something to criticize as a means of spurring him/her into action.

Initial prototypes were used as a means of eliciting knowledge from Dr Gulland. A simple decision tree was implemented in two separate prototypes PF1 and PF3, these respectively demonstrated strictly backward and strictly forward chaining strategies. In prototype PF1 backward chaining was demonstrated to show how it can be used to tightly tie questions to objectives. Discussion of these prototypes was used as the starting point for an informal interview. From this starting point Dr Gulland explained how, when looking at the management of a fishery, he would first request general information about the fishery, and then assessing this, would formulate a list of tasks that needed to be carried out.

The next stage was the development of a further prototype NPR.KB, with the aim of demonstrating how questions eliciting general information can be closely tied to stated objectives through a backward chaining implementation. A decision was made at this early stage that advice given to the user and the means to realize that advice should be separate. Thus 'plan of action' would be an incrementing list of textural instructions and 'plans' would be an incrementing list of procedural calls. These were both subdivided into three levels of priority.

EXPERTS INFERENCING STRATEGY

Dr Gulland did not like the backward chaining strategy as it did not reflect his own inferencing strategy. He expressed a preference to gather the whole range of information that might be required to make an assessment, before he related any of the information to any stated objective. A possible reason for using this strategy is that from a general description of the fishery, he could, based on his extensive experience, guess at the kind of objectives that were likely to become necessary in the future given the kind of biological, economic and industrial patterns that tended to occur in similar fisheries. Given the long time it takes to establish some fisheries management objectives e.g. a sampling programme and the long time it takes to accrue the necessary data, it seems from his advice, that often it is necessary to set certain tasks in motion because the objectives these served were likely to become necessities in the future, even if the user doesn't explicitly state them to be objectives.

ProTuna models this inferencing strategy and requires users to input a whole range of information regarding the fishery and state what their objectives in the management of the fishery might be. Careful analysis of the final rule base of Protuna shows that three out of five user stated objectives in Fisheries Sector Information Entry are not used in a single rule. For reasons mentioned above Dr Gulland attached a weak weighting to users explicitly stated objectives. In reality, as part of his inferencing strategy, he is saying :

(i) I am not really too worried what your stated goals are since they are unlikely to be realistic anyway.

(ii) Describe to me your general situation and from my experience I can tell you what your likely objectives will have to be.

(iii) And finally, based on these more likely objectives, I will tell you what you have to do to start preparing for and meeting them.

In relation to point (ii) above where ProTuna presents tasks in relation to Dr Gulland's supposed reformulated objectives, at no point are these made explicit so that the user can infer how these have been arrived at. This explains why there are a number of apparently arbitrary tasks that crop up in the concluding plans, although results from the tasks might be needed to meet the inferred objectives at no point in the process of consultation are these made explicit. This can be viewed as a design fault that can be improved.

In the newer version of the expert system ProFish, specific process have been incorporated to make the deep implicit knowledge more explicit for the user, this helping him to better understand the inferencing process. This has been achieved by providing rules for objectives and sub- division of objectives, so that objectives cannot clash and objectives become better defined. The aim is to help the user to be better able to define his objectives so that the expert system will become a more useful product.

KNOWLEDGE TRANSFER

In the development of further prototypes a pattern of knowledge transfer was adopted which could be carried out by correspondence. The pattern of knowledge representation and inference remained the same as before i.e. :

(i) ask lots of general questions.

(ii) then forward chain from this to things that need to be done

A resultant feature of this process is that it produced a 'flat' type of knowledge base. It made control of inference in the main national planner (ZPR.KB) relatively easy, but had the drawback that it was relatively easy in development to introduce inadvertently contradictions or inconsistencies into the rule base. As part of the development process the knowledge engineer drew the attention of the expert to any that occurred.

CONTROL OF INFERENCE

The inferencing process in the expert system is controlled using forward chaining, because of this some meta-level rules had to be introduced to control the flow of reasoning at the information gathering stage. For example, if the answer to 'Is there a foreign fleet?' was 'No.' then it isn't necessary to ask about foreign gear types. This would appear silly to the user and reduce their confidence in the system.

The introduction of meta-level knowledge had some unpredictable impacts. For example, if a question was not asked because it was illogical at the information gathering stage, a value for it was still required because the rule base was so flat. This happens as a consequence of the default backward chaining search strategy of the inference engine. It finds and activates the topic that asked the required question to supply the unknown value, this is again confusing to the user. As a means of preventing this happening the value of 'undefined' was assigned to all topics at the beginning. If a value is not assigned to topics as a consequence of meta-level control their values = 'undefined' and are processed in the intended manner at the evaluation stage.

As an alternative the same meta-level control could be imposed on the rules, but a similar problem would occur when values for topics are written to answer and log files. Formatting these files correctly is difficult enough without, the additional complication of having conditional statements contained in them as well, it is simpler to keep meta-level control to the information gathering stage. Then rules take care of themselves and values for 'undefined' in logfiles become self explanatory. One unforseen side effect of adopting this method is that it causes 'short-circuiting of the meta-level control when extensive use is made of the 'change answers' option.

TEXT HANDLING

Constant manipulation of the 'text strings' that represent the plan instructions in the system proved to be an unwieldily method of processing the rules. As a means of simplifying this each instruction was made into a value of a topic e.g. i1, i2, i3, ...i24. By this method only the topic name is manipulated, except when the actual instruction is written to the user or to a file, then the contents are retrieved. The 3 'poa' topics and the 3 'plans' topics become lists of instruction or plan headings, the contents under these headings are kept in separate topics named with those headings. An adaptation of this method could be used to programme in an explanation facility.

LOCATION IDENTIFIERS

Location identifiers provided as abbreviated codes and used as topic names within the programme are written into each window of the expert system. These were used during development to identify any areas of the programme that contained bugs or unclear questions. This procedural device enabled problems to be solved more rapidly as the knowledge base grew in size during development and prevent the need for laborious consultation each time the programme is run with few or no changes. These location identifiers are used to label instructions in the programme as well.

REVIEW AND RECALL OF ANSWERS

During development it was realised that it would be useful for the user to have a record of previous consultations. This was achieved by creating logfiles that recorded questions, users responses and advise given during each consultation. Each logfile is date and time stamped to provide the user with a unique identification of each consultation for future reference. Two types of logfiles have been incorporated into ProTuna to make it more user friendly and prevent repetitive cycles of use of the main ZPR knowledge base during consecutive consultations. In a practical fishery from one consultation to the next only a few answers would change their values. The answers.txt file enables the user to review and alter any answers after entering them, before they then proceed further forward to the evaluation stages. This removes the need to repeat a long cycle of repeated questions if the user has made one inadvertent entry during his consultation. The logfile.txt feature enables the user to save a set of answers and recall them for future use. A textfile when recalled can be reviewed and values altered as previously if required.

The answers.txt file is different to logfile.txt because the latter is annotated with additional text. In these features both saving and reading in is done using a loop counter to traverse a list in a topic called 'infolist', which contains all of the answers topic names. This list must be updated if any new questions and answers are introduced into the programme and the loop counter incremented accordingly. The ability to recall old answers proved useful during development,as it enabled interfacing with sub-module bases to be tested when answers in the parent ZPR.KB knowledge base were unchanging.

4 PROTUNA DESCRIPTION

INTRODUCTION

ProTuna is designed such that users react with the package as though they are discussing their problems and capabilities directly with a consultant, operating the system is known as a consultation. It is constructed as a set of separate modules that can run either in an integrated or independent fashion. Initially the user is asked through a set of Fishery Information Screens to respond to a number of basic questions. The answers to these questions are compiled into a profile of the characteristics of the fishery. These characteristics, which include biological, legal and administrative information are evaluated against the expert knowledge base that are built into the National Planner. proTuna responds by presenting a series of prioritised tasks (instructions) that must be preformed; these are usually the preparation of improved information for the fishery or other actions. Each of the tasks is followed up by a corresponding plan. The depth of each plan varies widely, in some circumstances it involves calling up and running another module in the system. For example, the Stock Assessment module, asks a further series of questions about the state of fishery statistics and responds with advice on whether there is sufficient information to undertake 'Production Modelling' to assess the abundance of fish stocks. Another series of question assesses whether the available data can be used for 'Length-based Assessment'. If the answers to each of these evaluate positively then a sub-module is run which performs this type of assessment.

SYSTEM DESCRIPTION

ProTuna a prototype Fisheries Management Expert System (FMES), specifically refers to tropical tuna fisheries, their characteristics, regulatory and legal environments, and administration in developing countries.

In ProTuna each module has been developed as a separate expert system. These are linked to the parent module when necessary with a load (*.ckb) command. Structuring ProTuna modularly simplified development and testing. It has also enabled useful features to be added gradually to a basic usable system. Developing a module for each specialised area allows different groups of people to proceed with development work, providing that the user interface is kept reasonably consistent.

Functionally ProTuna consists of a number of stages, these being :-

1. FISHERIES SECTOR INFORMATION ENTRY

   The user selects appropriate information from offered menu choices (selecting more than one where necessary), or enters information as prompted. This information can be considered to be a profile of the fisheries sector.

2. INFORMATION REVIEW

   The system compiles all selected and entered data into information screens for review. The user is given an opportunity to review and change each characteristic until satisfied that these represent an accurate summary of the fishery. If any changes to the answers are made, the altered summary can be re-written to the screen for final review.

3. EVALUATION

ProTuna evaluates the Fishery Sector Information against the rules and inferences - (the knowledge base) - built into the system and arrives at a composite conclusion for that consultation. At this stage the information is stored to a file answers.txt called ZPRANS.txt. This facility enables the user in subsequent consultations to utilise 'previous answers' and avoids the need to repeat inputting information each time a consultation is made.

4. TASKS

After evaluation of the Fishery Sector Information the system reaches a conclusion that identifies a series of tasks that need to be performed. These are shown on the screen under three priority levels, primary, secondary and tertiary tasks.

5. PLANS

Each identified task has a corresponding plan. Some of these are more complex than others. One, 'Plan 14' is a fish stock assessment planner which advises on different types of assessment that may be performed depending on the data available. These plans are shown on the screen with the same priority rating as the corresponding task priority.

PACKAGE HIERARCHY

The package comprises a number of separate but integrated sub-systems that are linked together in a hierarchical fashion. Figure 1 is a plan of programme modules and file structure. These modules are referred to on the opening screen menus as follows:

ZTF - **FISHERIES MANAGEMENT EXPERT SYSTEM** : This is the top level program that uses graphic identifiers of countries and can be used to call all other systems.

ZFS - **FRAME SURVEY** : This is an essential part of management planning but has not been fully developed for this release.

ZPR - **NATIONAL PLANNER** : The principal information and management expert system, utilising an expert knowledge base.

ZBP - **BIOPLANNER** : A linking module that offers two choices of stock assessment techniques - Production Modelling and Analytical Length-based Modelling.

ZPM - **PRODUCTION MODELLING** : Enables the user to assess the biological data available and whether it is sufficient and reliable enough to undertake stock assessment by production modelling. It does not undertake production modelling.

ZLM - **ANALYTICAL LENGTH-BASED MODELLING** : This system assesses biological data and its reliability to undertake analytical length-based modelling.[Note : LFDA (Length Frequency Distribution Analysis) is supplied along with this expert system. This package can be run independently of the expert system and comes supplied with its own user manual.]

ZSR - **SERVICES** : Enables the user at certain points in the hierarchy to exit to DOS, change Directory, examine a text file and check memory is available.

RUNPLANS **RUNNING THE SELECTED PLANS** : This system uses plans files created through the expert system evaluation at each of the three priority levels. The user can select which plan to run, identified by the corresponding task number and through an index. Most plans consist of text lists, the exception being plan 14 which loads and runs ZBP (BIOPLANNER) and subsequently ZPM and ZLM.

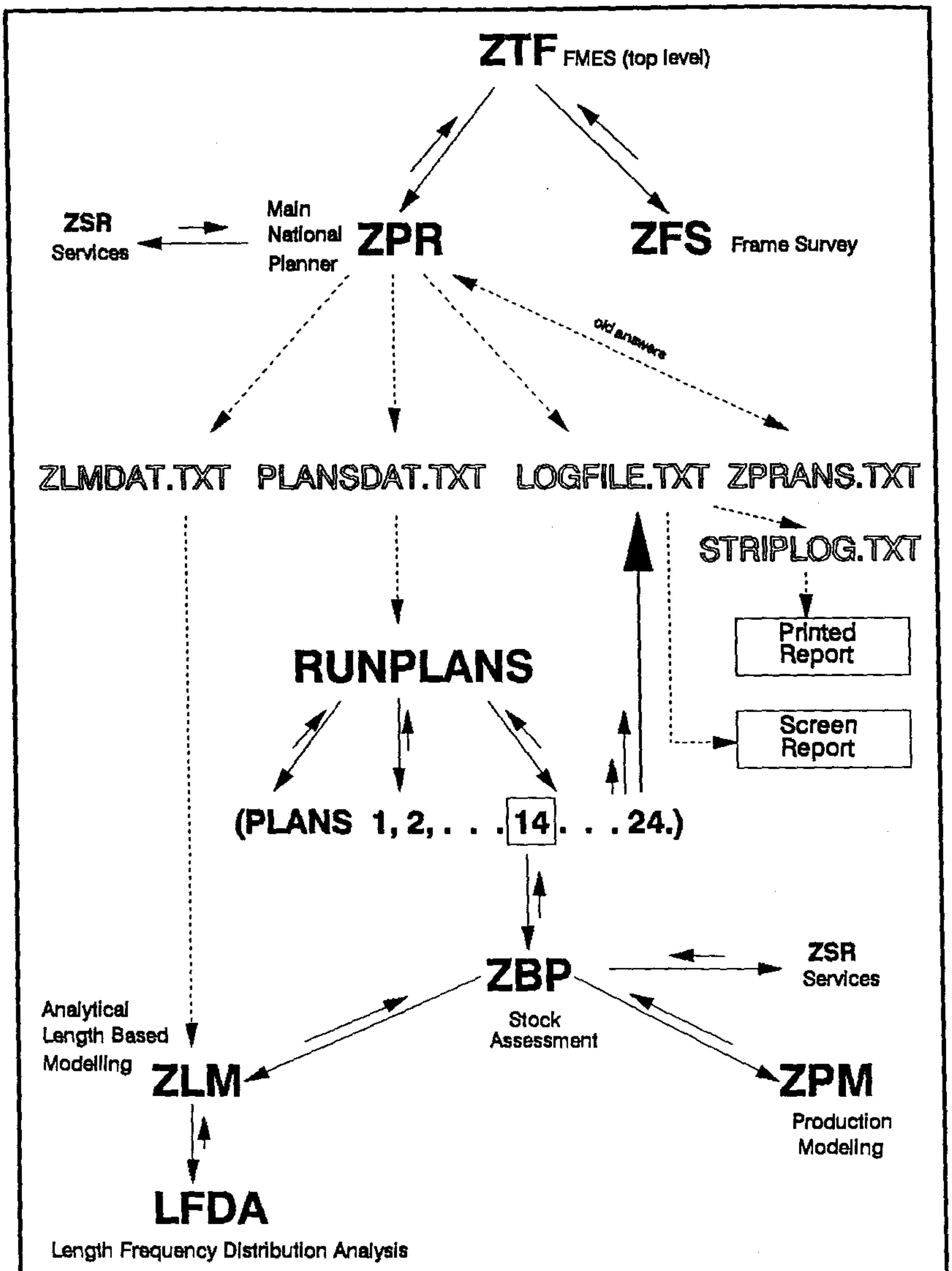

Figure 1
Plan of Program Modules & File Structure of the Fisheries Management Expert System.

ZPR	program module
ZLMDAT.TXT	data file
———▶	transfer of control
- - - - -▶	data transfer

DYNAMICALLY CREATED FILES

During each consultation four text files are produced by the system. These are used for various purposes as described below :

- **ZPRANS.txt** - This is produced by the system as a store of answers (selected and entered) from the previous consultation. Selecting 'old answers' at the beginning of a consultation enables the user to avoid repetitive use of the Fishery Sector Information Entry session by reading in the previous answers directly. These can be modified as required, prior to their evaluation.

- **LOGFILE.txt** - This is produced at user request following completion of the consultation. It contains full text of the user's answers plus the text of the conclusions in task priority. At the point of its production, a printed version can also be requested which will include date, time, user name and organisation. [It is highly recommended that such logfiles are stored in an orderly manner by adopting appropriate filenames following each consultation. LOGFILE.txt should be renamed, for example, to DATE.NAM (date.name) by the DOS rename facility.]

- **PLANSDAT.txt** - This is a file of all the plans associated with the identified tasks following the consultation. The sub-system RUNPLANS uses this file to present the plans to the screen. An option to print out each plan is available.

- **ZLMDAT.txt** - This is a data file that contains information to be used by the ZLM sub-system (length modelling) when it is run independently, ie the main system must be run at least once for ZLM to run. It contains species, foreign and national gear type information.

SUB-SYSTEM INDEPENDENCE

The following sub-systems can be run independently of the main system (ZTF).

- ZPR - NATIONAL PLANNER
- ZBP - BIOPLANNER
- ZPM - PRODUCTION MODELLING
- ZLM - ANALYTICAL LENGTH-BASED MODELLING

The sub-system ZLM can only be run independently if the main system ZPR has been run once in order to produce a data file, ZLMDAT.txt.

If sufficient memory is available on the particular computer that is in use, ZLM will also call up and run the LFDA (Length Frequency Distribution Analysis) package.

SUB-SYSTEM DEVELOPMENT DETAILS

Within the ProTuna there are 24 plans. After a consultation the user can select one or more plans corresponding to the tasks identified by the inferences and rules built into the system. The most elaborately developed is 'Plan 14', which is given the location identifier 'i14'. It is designed to satisfy instruction 14 which is "Initiate National Stock Assessment Programme". It links several sub-systems to the parent system, where appropriate, that will provide advice on the use of length based and production modelling methods for stock assessment, it is therefore essentially a sub-expert system.

PRODUCTION MODELLING SUB-SYSTEM (ZPM.KB)

This module advises users on the applicability of production modelling. Unlike the parent system ZPR a backward chaining strategy was incorporated. A conventional decision table was used to verify the logic of the sub-system and as the plan for testing the module. The 'geometric' nature of the logic allowed decision pathways to be used to validate the design with Dr. Gulland. The decision table (Table 1) identified two permutations where Dr Gulland had provided no recommended action for the user, these are marked *[1] and *[2] in Table 1. These were referred back to him for appropriate advice under such conditions. Extensive use was made of similar decision tables to validate other areas of ProTuna, usually subsets of rule bases.

TABLE 1. DECISION TABLE USED FOR VALIDATING LOGIC OF AND TESTING ZPM.KB (length based modelling expert system)

	RULES							
CONDITIONS	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
STATDATA	Y	Y	Y	Y	N	N	N	N
STATSTOCK	Y	Y	N	N	Y	Y	N	N
STATCHANGE	Y	N	Y	N	Y	N	Y	N
ACTIONS (conclusion gets)								
PRODDAT					*	*	*	*
PRODSTOCK			*	*			*	*
PRODCHANGE		*		*		*		*
PRODEXE	*							
PROD_TA_EXE		*						
PROD_TB_EXE			*[1]					
PROD_TC_EXE					*[2]			

The two permutations *[1] and *[2] when referred back to Dr Gulland were given the following appropriate advice.

*[1] Ask another question:

Is it likely that the change in effort on the rest of the stock are similar to changes in your area?

YES: Proceed with production modelling using caution.Results likely to be fairly accurate.

NO: Be very cautious. Changes in effort may not be reflected in changes in general stock abundance.

*[2] Not recommended that you proceed with production modelling.

Consider if data from other parts of stock may be used to give indication of changes in stock abundance.

The above decision table was implemented and tested in two different ways. Firstly as a series of eight separate rules, then the entire table was tested as a single rule with a series of nested if elses. The second construction ran slightly quicker and at first was considered to be the more elegant alternative. Both when tested completely for **ALL** permutations of conditions performed correctly. With the nested construction it is difficult to divulge the logic evolved.

The use of separate rules would make maintenance and updating easier however each of the conditions needs to be reevaluated for each rule, even though the condition has not changed. Figure 2 shows the ZPM.KB production modelling expert system decision pathway.

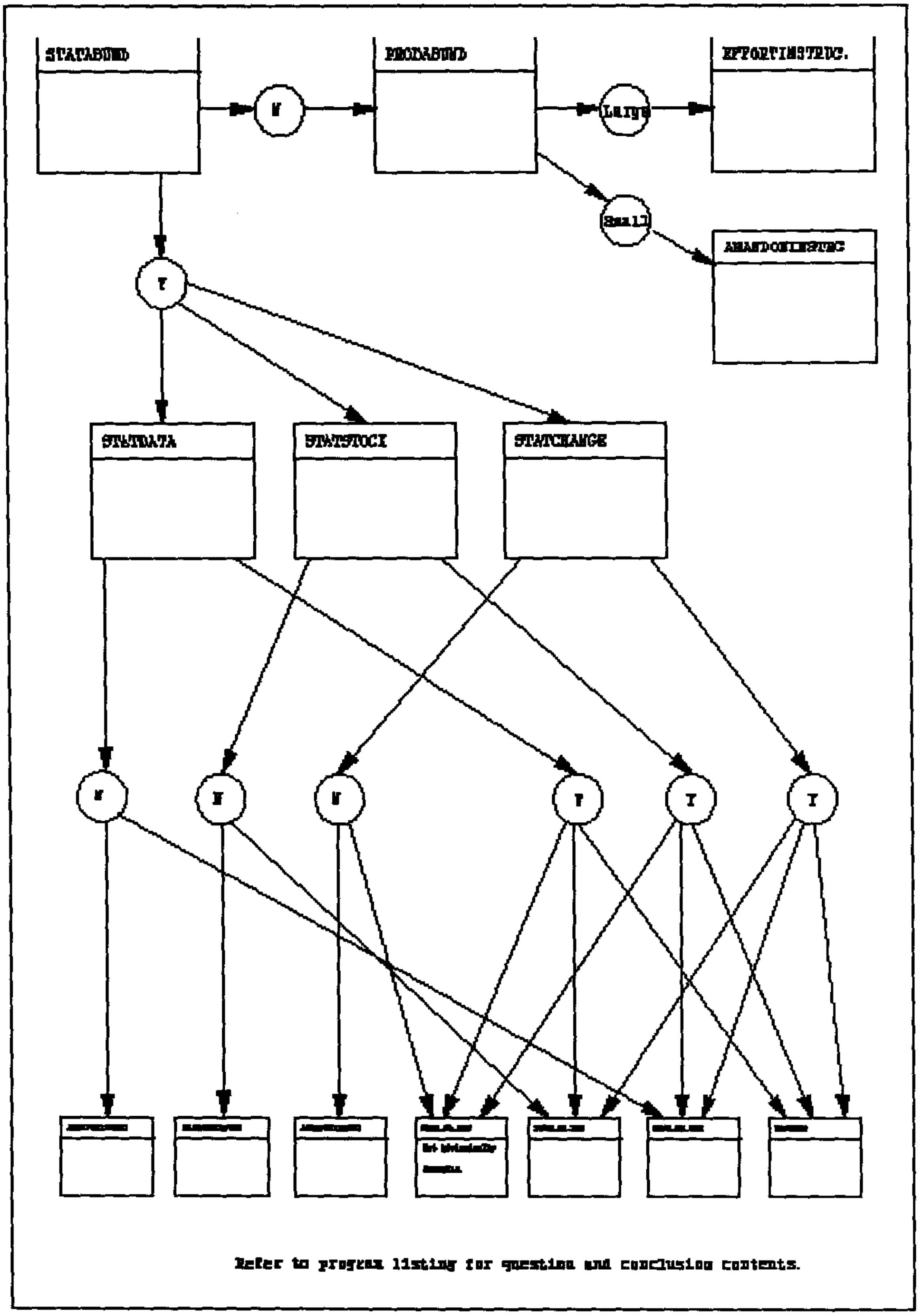

Figure 2. ZPM.KB Production Modelling Expert System Decision Pathway

LENGTH MODELLING SUB-SYSTEM (ZLM.KB)

This module asks the user for information about general patterns in their length frequency data and uses this in conjunction with values for topics from the parent module, (ZPR.KB) national planner knowledge base, obtained by 'blackboarding' i.e. reading in the values from a common data file, ZPR.ANS. This data file has to be present and contain the relative data for ZLM.KB to run.

Additional features offer the user increased flexibility. As an alternative to being irrevocably channelled down a given decision pathway, the user can make choices from a menu after the module has advised them of the pathway they should follow, from the information available. This facility enables the user to test his own 'what if' scenarios. A greater use of Hypertext allows the system to provide more advise to the user when choices are presented. A user can run an option through hypertext instead of selection from menu choice, offering a more natural way of selecting options than through hypertext alone. If this method is used careful control of structures and transfer of control between topics is required to prevent 'spaghetti situations' occurring.

Two of the final options in ZLM.KB attempt to load LFDA.EXE. This is a separate pascal programme supplied together with ProTuna. It offers a toolbox of methods for estimating population parameters from length measurement data. The LFDA programme requires P.C.'s with extended RAM to overcome space constraints when an attempt is made to run it. When it was realised during development that most machines would have a space constraint, memory management was improved to detect any conflict before it occurred. The programme informs the user if insufficient memory is available and advises that LFDA is run separately later.

IMPROVING ON PROTUNA

Evaluation of ProTuna both in house and in field tests demonstrated a number of drawbacks.

A Its structure is rigid and imposes unnecessary constraints on the development of management measures. There are a number of reasons for this :

- (i) It attempts to do everything in one go, from ascertaining the local situation that prevails to begin with, through to running assessment programmes. It is obviously unrealistic to attempt this in real time because it takes time to organize and run both information collection (sampling programmes and collation of catch data) and information processing (fisheries assessment calculations).

- (ii) It 'dives in at the deep end' immediately asking for specific objectives and ever finer details of information as it progresses down a rigidly defined decision pathway. If at any point a user cannot answer or supply the required information it is not possible to progress further.

B The rule base in the main parent module ('ZPR' national planner) is poorly structured, the rules have been created in isolation and are 'flat'. The rule base needs to be :

- (i) Increased in its scope.
- (ii) Made syntactically simpler.
- (iii) Have its rules separated out into a number of appropriate 'stages'.
- (iv) Be more closely coupled to other rules through the use of backward chaining, use of intermediate rules and referral to the 'log file'.

In ProTuna two attempts were made to improve flexibility. Firstly, responses from a run of the 'ZPR' (National Planner) parent system can be stored so that they can be re-run, see the previous section on review and recall of answers. Also in the sub-system 'ZLM' (Length Based Modelling) menu choices are provided that allow the user to choose a pathway to follow after considering advice, rather than being rigidly guided down a pathway.

Considerable thought was given to the advisability of attempting to run all programmes at once. Apart from memory constraints, several of the information collection programmes recommended take considerable time to complete. This lead to the idea of developing an expert system that could be split into 'stages', each of these 'blackboarding' relevant information with other stages via files. This system would allow estimates of parameters made in external systems to be passed back into the expert system and used in further evaluation. ProFish incorporates these ideas, as an expert system it co-ordinates with other modules via knowledgeable decisions as part of its expert role

PROFISH

A second product called ProFish was developed with the aim of improving on ProTuna as outlined above and to provide a broader domain to cover all types of fisheries in developing countries.

With limited resources it was not feasible to implement some of the more rigorous methods considered e.g. those based around Shepherd's ideas (Shepherd Chapter 2 in Gulland 1988) using detailed systems analysis and design. A theoretical assessment of the potential of these methods is presented in Appendix 2.

Profish was designed to be better suited to the working practises of fisheries managers. The most important decisions and advice made within the system are dependent on the user providing reasonable yield estimates. Therefore a separate Information and Research Planner first offers advice on methods of making such estimates and whether production models or analytical methods could offer the best approach.

The next three 'modules' gradually build up a comprehensive picture of the fishery and its characteristics. The fourth module then carefully aids the manager to a realistic choice of management objectives. Based on this and all of the information asked in the first three stages, various management measures are suggested at three priority levels. The fifth and final module allows the user to chose each of the suggested management measures for enlargement of advice.

Though the main five modules should be used as a series the user can leave and return at any level and separate consultations can be run in parallel e.g. to consider the effects of varying current management measures or objectives.

Each level communicates with the next via a series of data files. The answers for a given stage can also be saved to 'old answers files', the user can call these in for reference during future consultations.

A full logfile is appended at each stage providing the user with a complete record of his/her consultations.

File handling facilities have been much improved giving greater convenience and flexibility to the fisheries manager. The consultation process can now be realistically run as a series of stages allowing the user a natural break in order to run or rerun a given stage once associated external activities have been completed.

The reader is referred to the ProFish user manual for a detailed description of the structure and functions of ProFish.

5 CONCLUSIONS AND RECOMMENDATIONS

SHORT TERM

Profish is a practically useful and fully working product. The carefully designed modular and time staged structure is intended to be and offers an excellent platform for further expansion in future as needs are perceived and if and when resources become available to meet them. New modules can be added and existing ones expanded and improved.

LONG TERM

Further study is required into management as control of trends, optimization theory, graph theory and the use of structured objects. Greater practical experience can be gained by attempting to produce prototypes based on structured objects. If time and resources allow, such work would complement the experience gained in using production rule systems and would in the long term allow a more thorough evaluation of artificial intelligence for use in fisheries.

There is scope for exploring the use of 'Expert Database Systems' which can serve to :

(i) Make more direct use of fisheries data, parameter estimation and modelling software.

(ii) Improve the design of systems by utilising normal Database Management System functions and structures where expertise is not involved.

A thorough design proposal based on these ideas is presented in Annex 2. It is based on the framework proposed by Shepherd (Chapter 2 in Gulland 1988), for management goal definition in relation to the 'dependence of assessments on information' and the 'information content of data'.

REFERENCES

*** Anon.** 1987 CATCUV 1: Fishery Management Expert System. Proceedings of ORIA '87: Artificial Intelligence and the Sea. Inst. Robotique & Intelligence Artificielle, Marseille, France.

Arang, A.A. & Golden, B.L. 1987. Expert Systems, Microcomputers and Operations Research. Working Paper No. 7. University of Maryland.

Benson, I. (Ed). 1986. Intelligent Machinery. Theory and Practise. C.U.P 1987.

Berry, D.C. 1987. The Problem of Implicit Knowledge. Expert Systems Vol 4. No 3. Aug 1987. Learned Information (Pub).

*** Bielawski, L. & Lewand, R.** 1989. REGIS: Regional Information System for African Aquaculture. F.A.O. and National Agricultural Library, USA Dept. of Agriculture.

*** Borch, O.J., Hartvigsen, G.** 1990. STRATEX: A Knowledge Based System for Strategic Market Planning in Small Firms (fish export). A.I. Communications, March 1990 vol 3, no 1, pp 12-24.

Cohen, C. 1989. Merging Expert Systems and Databases. A.I. Expert. Feb 1989.

Doulman, D.J. (Ed.) 1987 Tuna Issues and Perspectives in the Pacific Islands Region. East-West Centre 1987.

Dowsing, R.D., Rayward-Smith, V.J., Walter, C.D. 1986. A first Course in Formal Logic and its Applications in Computer Science. Blackwell Scientific Publications.

Eliot, L.B. 1987. Investigating the Nature of Expertise. Expert Systems Vol 4. No 3. Aug 1987. Learned Information (Pub.).

Fiderio, J.,Frisse, M.,Begeman, M.L. & Conklin, J. 1988. Hypertext Review. Byte. Oct 1988.

Finkelstein, R. & Pascal, F. 1988. SQL Database Management Systems. A group review. Byte. Jan 1988.

Forsyth, R. Expert Systems, Principals and Case Studies. 2nd Edition 1989.
Chapman and Hall Computing

*** Froese, R., Schofer, W.** 1987. Computer Aided Identification of Fish Larvae. ICES, Copenhagen (Denmark) 1987. 10pp.

*** Golden, B.L. & Rothschild, B.J.** 1987. A Microcomputer Based Decision Support System For Multispecies Fishery Management. Technical Report No. 1. The University of Maryland.

Gulland, J.A. 1985. Fish Stock Assessment. A Manual Of Basic Methods. FAO / Wiley series on food and agriculture

Gulland, J.A. (Ed) Fish Population Dynamics. 2nd edition (1988)
Wiley - Interscience

Gulland, J.A. & Rosenberg, A.A. Manual of Length Based Methods. In press.

Guifoyle, C. 1987. Inside the Engine. Expert System User. Feb 1987.

Jackson, P.J. 1986. Introduction to Expert Systems. Addison-Wesley Ltd.

Jinman, R. 1989. Expert System Overview. Personal Computer Magazine. May 1989.

* **Jones, J.C.** 1989. Expert Systems for Fisheries Management. Biological Computation, University of York. R.R.A.G.

Johnson, P.E., Nachstein C.J. & Zualkeman, I.A. 1987. Consultant Expertise. Expert Systems. Vol 4. No 3. Aug 1987. Learned Information (Pub.).

Joseph, J., Klawe, W. & Murphy, P. 1988 Tuna and Billfish. Inter American Tropical Tuna Commission.

Layzell, P.J. & Loucopoulos, P. 1987. Systems Analysis and Development. 2nd ed. Chartwell-Bratt.

Le Gall, J.Y. 1989. Teledetection Satellitaire et Pecheries Thonieres Oceaniques. United Nations Food and Agriculture Organization, Rome, 1989.

Olson, J.R. & Reuter, H.H. 1987. Methods for Knowledge Acquisition. Expert Systems Vol 4. No 3. Aug 1987. Learned Information. (Pub.).

* **Petit, M. & Stretta, J.M.** 1989. Two Tools Available for an Operational Tuna Fishery: Aerospatial Teledetection and Expert Systems Models. CICAA vol 30, no 2, pp 500-505

Raucher, H.M. 1987. Expert Systems for Natural Resource Management. The Compiler Nov.Dec. 1987.

Rosenberg, A.A. 1989. ODA Project Proposal. 'The Development of Computer Aids for Fish Stock Assessment Methods and Management Policy Development for the Island states of the South West Pacific and Indian Ocean.'

Rosenberg, A.A. 1988. Fishery Data Collection Systems for Eastern Caribbean Islands. OECS fishery report no. 2.

Rowley, D.T. 1990. Review of B.E.A.G.L.E., ('Biological Evolutionary Algorithm Generating Logical Expressions.'). Expert Systems. Vol 7. No 1. Feb 1990. Learned Information (Pub).

* **Ryan, J.D. & Smith, P.E.** 1985. An expert system for fisheries management. in Oceans '85. Proceedings: Ocean Engineering and the Environment. Vol 2, pp. 1114-1117 San Diego CA. USA. Nov. 1985.

Shepherd, J.G. 1988. Fish stock assessments and their data requirements. Chapt 2 in Fish Population Dynamics. J.A. Gulland Ed. Wiley 1988.

Shepherd, J.G. 1987. A weakly parametric method for estimating growth parameters from length composition data. p.113-119. In D. Pauly and G.R. Morgan (eds.) Length-based methods in fisheries research. ICLARM Conference Proceedings 13, 468p.

Shepherd, J.G. 1987. Towards a method for short-term forecasting of catch rates based on length compositions. p.167-176. In D. Pauly and G.R. Morgan (eds.) Length-based methods in fisheries research. ICLARM Conference Proceedings 13, 468p.

Smith, J.M. 1986. Expert Database Systems. A Database perspective. In proceedings from the first international workshop on Expert Database Systems. (Kerschberg L. Ed.) Benjamin / Cummings 1986.

Sommerville, I. 1985. Software Engineering. 2nd ed. Addison-Wesley.

* **Steinbach, C., Pereira, C.,** 1989. SALMEX: An Intelligent Knowledge Based System for the Diagnosis of Salmonoid Fish Diseases. Proceedings of the Scandinavian Conference on Artificial Intelligence 1989.

Swartont, W.R. & Smolinar, S.W. 1987. On Making Expert Systems More Like Experts. Expert Systems. Vol 4. No 3. Aug 1987. Learned Information. (Pub).

Swaffield, G. & Knight, B. 1990. Systems Analysis For Knowledge Elicitation. Expert Systems. Vol 7. No 2. May 1990. Learned Information (Pub.).

Vedder, R.G. 1989. P.C.-based expert system shells: some desirable and less desirable characteristics. In Expert Systems, Feb 1989, Vol 6 No 1.

Whitehorn, M. 1989. The Key to Expert Knowledge. An appreciation of Hypertext. P.C. User 1 - 14 March 1989.

Whittington, R.P. 1988. Database Systems Engineering. Oxford University Press.

Yazdani, M. 1989. Shells versus Toolkits. Expert System User. Jan 1989.