DAIRYMAP: A WEB-BASED EXPERT SYSTEM FOR DAIRY HERD MANAGEMENT

by

SANJAY S. CHELLAPILLA

(Under the Direction of WALTER D. POTTER)

ABSTRACT

This thesis describes the design and implementation of DairyMAP, a Web-based benchmarking analysis and Expert System for Dairy Herd Producers, as part of the Dairy Management Analysis Program undertaken by the Edgar L. Rhodes Center for Animal and Dairy Science. The system consists of two major components – a preliminary statistical benchmarking analysis (based on Dairy Herd Information reports provided by the Dairy Records Management Systems, Inc., in Raleigh, NC) and, a detailed expert evaluation of the four major areas of dairy herd management, *viz.,* Somatic Cell Count and Mastitis, Reproduction, Genetics, and Milk Production. The preliminary analysis provides information to the producer about the areas of concern within each component of dairy management, and suggests further evaluation and diagnosis by the Expert System, concluding with comments and recommendations for improving the producer's herd. The entire system is Web-based, allowing a producer from anywhere in the United States to utilize the system. Though there has been similar work with Expert Systems for individual areas in Dairy Management, this project aims to provide a single integrated system evaluating all the four major areas, with the preliminary analysis limited only by the dairy herd data available. This report describes the system as operational at the end of the first phase of the project, with work ongoing in improving the system and adding features to the system. The system is available at the URL `http://dairymap.ads.uga.edu/`

INDEX WORDS: Dairy Herd Management, Expert System, World Wide Web, WWW, DairyMAP, Knowledge-based system, Decision Support System, decision making

DAIRYMAP: A WEB-BASED EXPERT SYSTEM FOR DAIRY HERD MANAGEMENT

by

SANJAY S. CHELLAPILLA

B.Tech., Andhra University, India, 1995

M.Tech., Indian Institute of Technology, Kanpur, India, 1999

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2003

DAIRYMAP: A WEB-BASED EXPERT SYSTEM FOR DAIRY HERD

MANAGEMENT

by

SANJAY S. CHELLAPILLA

Major Professor: Walter D. Potter

Committee: Khaled Rasheed
James W. Smith

Electronic Version Approved:

Maureen Grasso
Dean of the Graduate School
The University of Georgia
August 2003

for

Amma, Deepu, and all my wonderful family

ACKNOWLEDGEMENTS

I thank my advisor Dr. Potter for his patience, continued support and encouragement. I also thank Drs. Smith, Ely, Gilson and Graves for their patient support, guidance and encouragement, and gratefully acknowledge the continued financial support by the Animal and Dairy Science department which allowed me to focus on my research and studies. I thank Dr. Rasheed for serving on my committee.

Thanks to Sivarajah for telling me about the AI program. Friends, colleagues, and Angie Paul at the AI Center, Swarna and family, Anand and family, and numerous people (both on and off campus).

Rajan, Padmakar, Kedar, Biplab, the *ctop96* folks, and all the “kalianpur bakaits” from IITK. Finally, the University, with all its academic and extra-curricular programs and resources, town-and-gown with the “Classic City” of Athens, which has made for a growing and enriching experience.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Problem-solving is a classical attribute of intelligence, human or artificial. Reasoning with logic using domain-specific knowledge forms an important part of the problem-solving strategy for real-world problems. Expert knowledge and reasoning consists of a theoretical understanding of the domain, with facts, heuristics and rules specific to that domain. Such domain-specific knowledge (heuristics and facts) is contained in a knowledge-base, whether in humans or computers. One successful application of reasoning techniques in Artificial Intelligence (AI) using facts and rules is in building computational systems which reason over such knowledge-bases, known as Expert Systems. In helping human beings to make decisions, they are also known as Decision Support Systems. They have found application in real-world problems in many domains, such as medical diagnosis, mining and prospecting, discovering molecular structure (to name a few popular examples). A survey of expert systems in use (circa 1988) is given in [Feigenbaum, McCorduck and Nii 1988]. One definition of an Expert System as defined in [Feigenbaum, McCorduck and Nii 1988] is:

> …Often, the term expert systems is reserved for programs whose knowledge base contains the knowledge used by human experts, in contrast to knowledge gathered from textbooks or non-experts. More often than not, the two terms – expert system and knowledge-based system – are often used synonymously…

The advantages in using Expert Systems cannot be overemphasized. In containing and utilizing human expert domain-knowledge, they can perform the role of experts in that specific domain, and prove helpful to experts and non-experts alike. The human expert, in being aided by such decision support systems, can relegate well-known inference procedures and solutions to the automated system, thus being able to focus on further investigation where needed. At the same time, non-experts with problems that readily yield themselves to established diagnostic processes and solutions can be helped quickly without the need for human intervention. Another advantage lies in the scope for bringing in objectivity in such cases seeking expert help.

1.1 Background and related work

Agriculture, and dairy herd management in particular, as human endeavors in specialized domains, require extensive knowledge and human expertise in solving problems, and as such, has been the focus of several efforts in automation with AI and Expert Systems. An overview of Expert Systems in Agriculture is given in [McKinion and Lemmon 1985a]. Examples include farm machinery selection [Parmar et al. 1996], COMAX – a cotton crop management expert system [McKinnon and Lemmon 1985b, Lemmon 1986], weed identification in turf-grass management [Fermanian et al. 1985], crop insurance strategies [Helms et al. 1990], PLANT/ds – an expert system to diagnose soybean diseases [Michalski et al. 1983], and, XLAYER – a poultry layer management advisory expert system [Schmisseur and Pankratz 1989].

The importance of automation of dairy herd management to farmers, herd producers and dairy extension educators cannot be overemphasized. There have hence been several efforts in this direction. A lactation curve analysis program [Fourdraine et al. 1990], a dairy control and management system in automated milking farms [Devir et al. 1993], and a program for the use of prostaglandin in dairy herd management

[Heueriser et al. 1993] are a few examples. Dairy herd management, along with requiring extensive background knowledge and heuristics, is data-intensive. An automated reasoning system which manipulates and infers over such data offers numerous benefits to users interested in improving herd performance. The area of dairy management is well-suited for applying expert systems since many factors need to be considered by the expert in formulating a decision. Expert systems choose the appropriate rules as needed, and new rules can be added to the ruleset as domain expert knowledge is engineered and encoded.

A few dairy management Expert Systems discussed in the literature include: a diagnostic aid for reproduction problems in dairy cattle [Levins and Varner 1987], expert systems for culling management of beef cows [Oltjen et al.1987], MAST – a data-driven expert system for Mastitis [Allore et. al 1992], an expert system for evaluation of reproductive performance [Demecq et al. 1991], DXMAS [Schmisseur and Gamroth 1993], which provides management advice by analyzing economic and production performance data provided by dairy operators, LAIT-XPERT VACHES [Pellerin et al. 1994], an expert system which calculates objectives in milk production, fat and protein production, reproduction and other areas of the dairy enterprise, and DAIRYPRO [Kerr et al. 1999], a knowledge-based decision support system to help northern Australian dairy farmers make decisions on the general level of efficiency of their farms.

1.2 The Dairy Management Analysis Program (DairyMAP)

Dairy production is an area where many interacting factors determine the degree to which a dairy farm is considered productive. Dairy experts typically meet with producers, analyze the various aspects of herd management and suggest recommendations for improvement. Given dairy production information for a farm, experts first perform a statistical benchmarking analysis, which compares production levels of the farm with those averaged from farms all over the country, and identify areas of concern. The expert then suggests improvements within each area of concern in a detailed analysis relying on his/her expert knowledge of the domain. The many interacting factors affecting dairy production do not yield merely to statistical analysis, and their non-linear relationships present a challenging problem for analysis by dairy experts. Also, given that the analysis depends on a human expert, different approaches lead to varied conclusions. Hence, automating the process would be a boon to a herd producer in both ways – (i) a numeric benchmarking of the herd identifying areas of concern, and, (ii) a consistent, objective, detailed expert analysis recommending ways to remedy the problems with the herd.

The University of Georgia Extension Service has several programs in helping dairy farmers and herd producers improve the yields of their herds. One such program is the Dairy Management Analysis Program, or DairyMAP. Dairy educators and experts (such as those at the University) utilize herd summary data from the Dairy Records Management Systems Inc., Raleigh NC, in analyzing the quality of a producer's herd by statistical analysis, benchmarking, and providing recommendations for improvement. A background of DairyMAP is discussed in [Smith 1992], [Smith 1993], [Smith et al. 1995], and [Smith and Rodekohr 1996].

This thesis discusses the development of a World-Wide-Web (WWW) accessible, statistical benchmarking analysis and rule-based Expert System to help dairy producers improve the quality of their herds and yields. The system being WWW-accessible affords ease of availability and convenience to a dairy producer located anywhere in the country.

The benchmarking analysis evaluates the standing of a herd as compared to herds of a similar size within the same geographical region. This is done by utilizing the data available from a database of herds country-wide. The database allows a benchmarking analysis of a producer's herd in terms of various dairy management parameters like milk production yields and costs, genetic characteristics, reproduction, and a common disease called Mastitis which manifests itself as an increase in the Somatic cell count. Thus, this system addresses all the four areas (viz., Somatic Cell Count/Mastitis, Milk production, Reproduction and Genetics) important to holistic dairy management, in detail.

A herd producer seeking expert help typically meets with dairy experts who utilize their domain expertise to ask questions and recommend procedures for improvement. Hence, a key component in developing an expert system is knowledge engineering (wherein domain expert knowledge is formally gathered), and knowledge representation and encoding, to manipulate the knowledge base to form inferences. In DairyMAP, the knowledge experts are a group of Dairy Extension experts at the University (Professors Ely, Gilson, Smith and Graves), who elucidated the typical question-answer diagnostic process in the form of clear, sequenced, questions and answers. The questions in the sequence would also change depending on the answers given by the user. It hence became easier to encode this process in terms of the conventional 'If-Then' rules. The system maintains a track of each individual session (since it is WWW-accessible) so that simultaneous sessions by unique users do not interfere. The user would enter his/her herd data from a standard report, and, after a benchmarking analysis, obtain preliminary summaries informing him/her of the areas of concern within an area of dairy management. The user can then choose to obtain further expert guidance by proceeding to the expert system. The expert system would diagnose the problems by asking questions and then suggest appropriate measures to follow. The following chapters discuss the design and implementation of the preliminary benchmarking analysis and the expert system, concluding with a summary and scope for further work. Chapter 2 discusses the preliminary benchmarking analysis, wherein data about a producer's herd is compared with herds of a similar size within the same region (as the dairy producer is in) in the United States. The percentile standing of the herd, ratings tables and summaries with areas of concern are displayed for each of the four aspects of dairy management. At the end of this analysis, the producer has a choice on whether to proceed to a detailed expert analysis (in a question-answer session) which is the core knowledge-based expert system, concluding with recommendations and suggestions for improvement (Chapter 3). Chapter 4 concludes with a summary and scope for further work.

CHAPTER 2

PRELIMINARY BENCHMARKING ANALYSIS: DESIGN AND IMPLEMENTATION

This chapter discusses the design and implementation of the preliminary benchmarking analysis of a producer's herd.

2.1 Design

Dairy producers who are members of the Dairy Herd Improvement Association (DHIA) have access to a database of dairy production information containing records of herd performance data. The data include measures of milk production efficiency, herd health, yields, and so on. These records are available from the Dairy Records Management Systems, Inc., Raleigh, NC. The report format that is used by dairy experts is the DHI-202 report [DHI202]. The database consists of herds classified into four major geographical regions *viz.,* South, Mid-South, North-East, and Mid-West. The producer's herd is benchmarked against herds of a similar size within the same geographical region. The analysis consists of the following stages

- Welcome page: this page welcomes a user to the DairyMAP and is the entry page to the system. A user needs to be registered (at no charge) with a user-name and a password before he/she can proceed further. A new (unregistered) user is allowed to sign-up to use the system through a hyperlink to the registration page. After registration the user is directed to the welcome/login page to access the system. A hyperlink leading to further helpful information on using the system is also available. Upon registration, a confirmation email is sent to the user at his/her chosen email address.

- General Herd Information: a data-entry page requiring input of the name of the producer, herd identification number, location (name of state), and size of herd.
- Herd Summary Information: a data-entry page requiring input of various performance values of the producer's herd as listed in the DHI-202 summary report, totaling thirty-five in number. Numbers in parentheses beside the labeled data input boxes on the page point to the location on a sample DHI-202 report where a producer can find the corresponding data value. A link to the sample DHI-202 report with numbered fields helps a producer to find the required information.
- Dairy Herd Management Evaluation: an output page displaying a comparison of the producer's herd information with herd information of a similar sized herd within the same geographical region. The display page also highlights values of benchmark data closest to those items of the producer's herd, and contains buttons for detailed evaluations.
- Detailed Evaluations: a series of pages requiring user input, and output pages displaying ratings and summaries of herd performance and areas of concern within each major area of interest *viz.,* Somatic Cell Count score (for Mastitis evaluation), Milk Production, Reproduction, and Genetics. The summary pages at the end of each detailed evaluation allow the producer to either (a) return to the Herd Management Evaluation, and hence, on to the next area of detailed evaluation, (b) continue to the detailed Expert System analysis in the chosen area

of interest, or (c) exit the analysis and proceed to a system evaluation page which requests user feedback, bug reports and comments on the system. Currently, each of the major areas of interest, *viz.,* Somatic Cell Count (Mastitis), Milk Production, Genetics, and Reproduction, have a corresponding Expert System entry page. The system evaluation page is a simple questionnaire allowing users to provide feedback regarding usability, navigation, comments and/or bug reports if any. The answers are emailed to the maintainers of the system for their notice, and remedial action.

A security feature of the system is that of session-expiration wherein a user's session expires (requiring re-authentication by login) on a time-period of inactivity on the part of the user (adjustable by the system administrator). The flow of control between the stages of the preliminary statistical analysis is illustrated in the following pages. The dotted arrows show possible alternative paths of execution during the analysis.

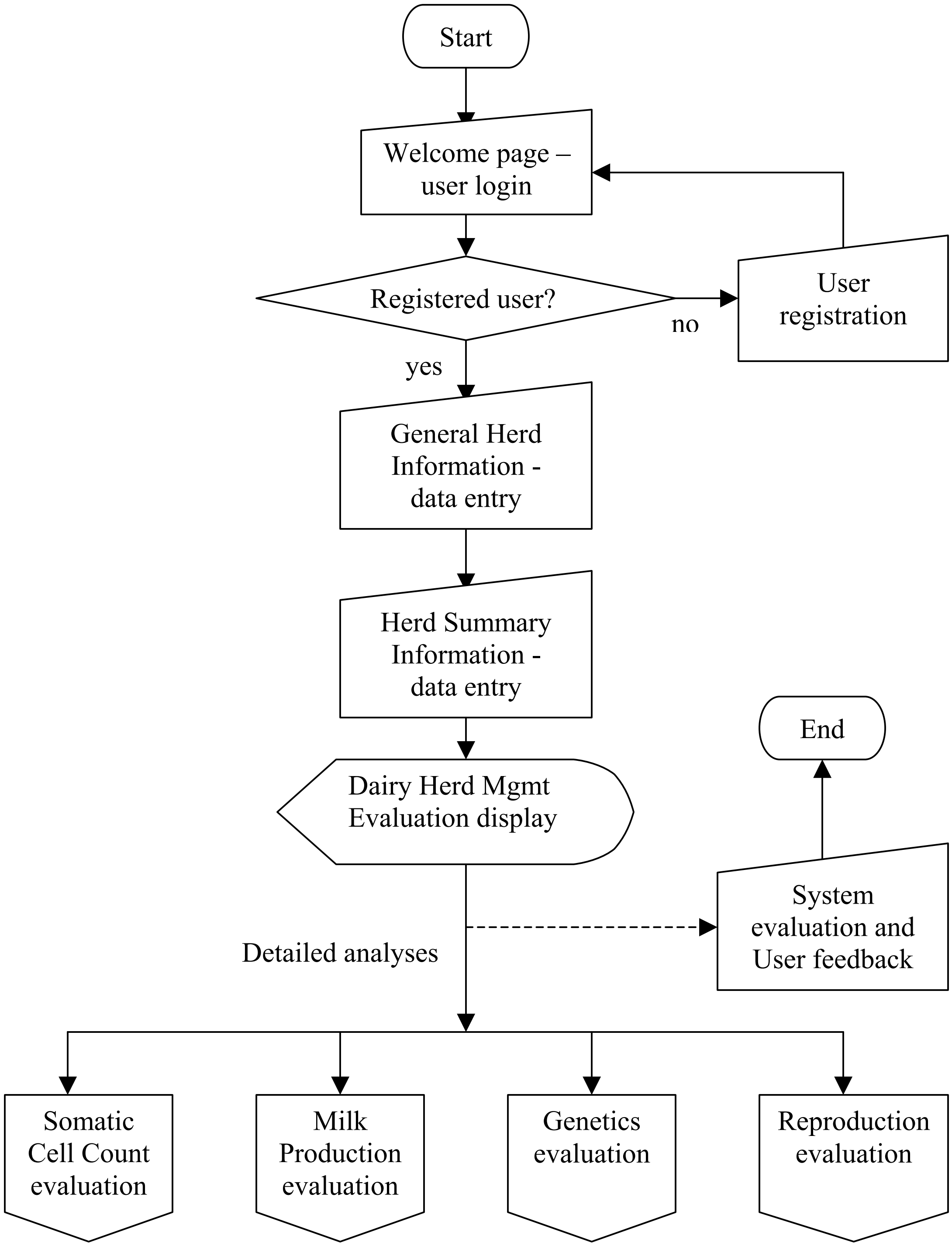


Figure 2.1: Preliminary benchmarking analysis

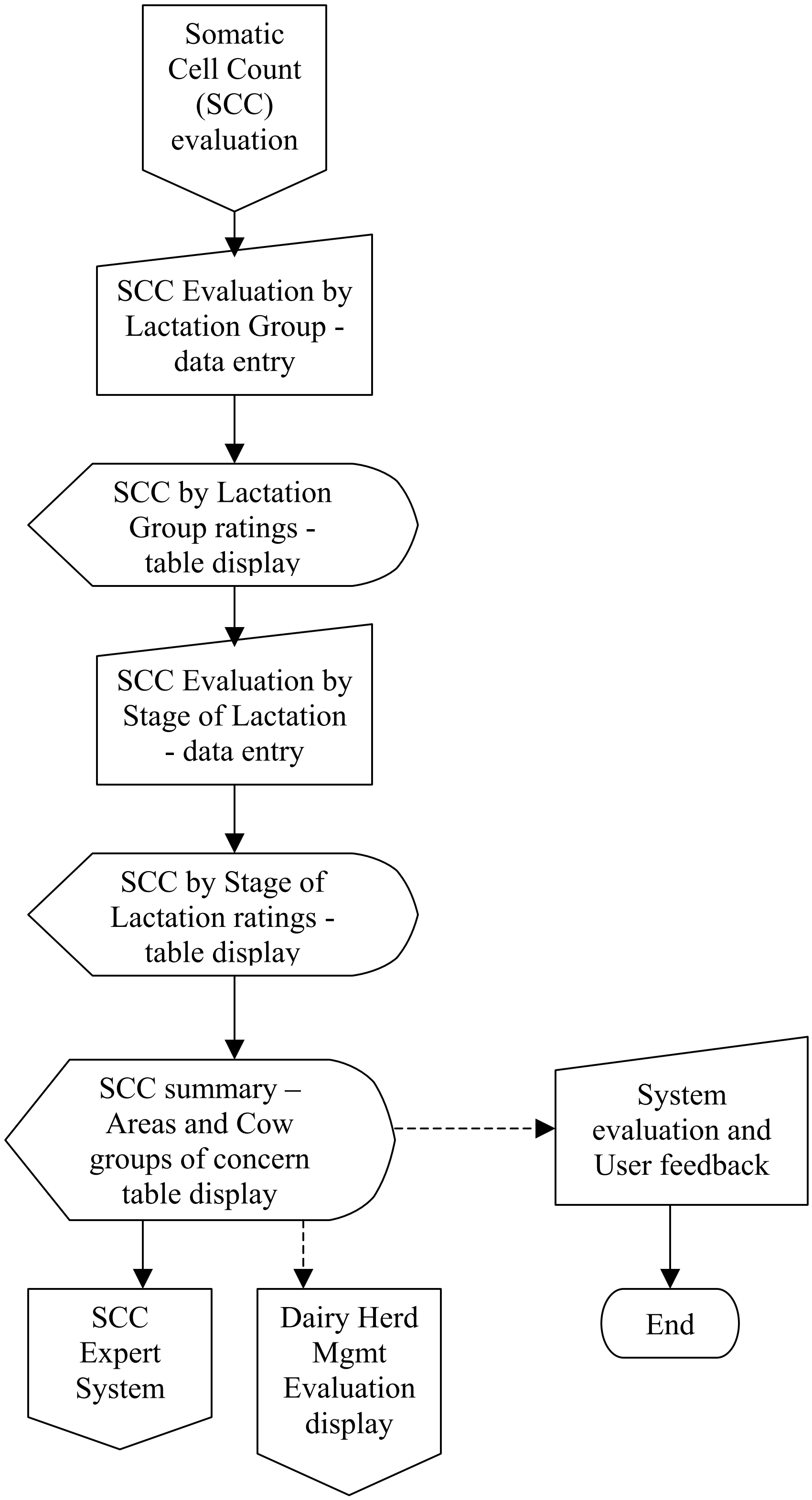


Figure 2.2: Detailed evaluation: Somatic Cell Count (SCC) analysis

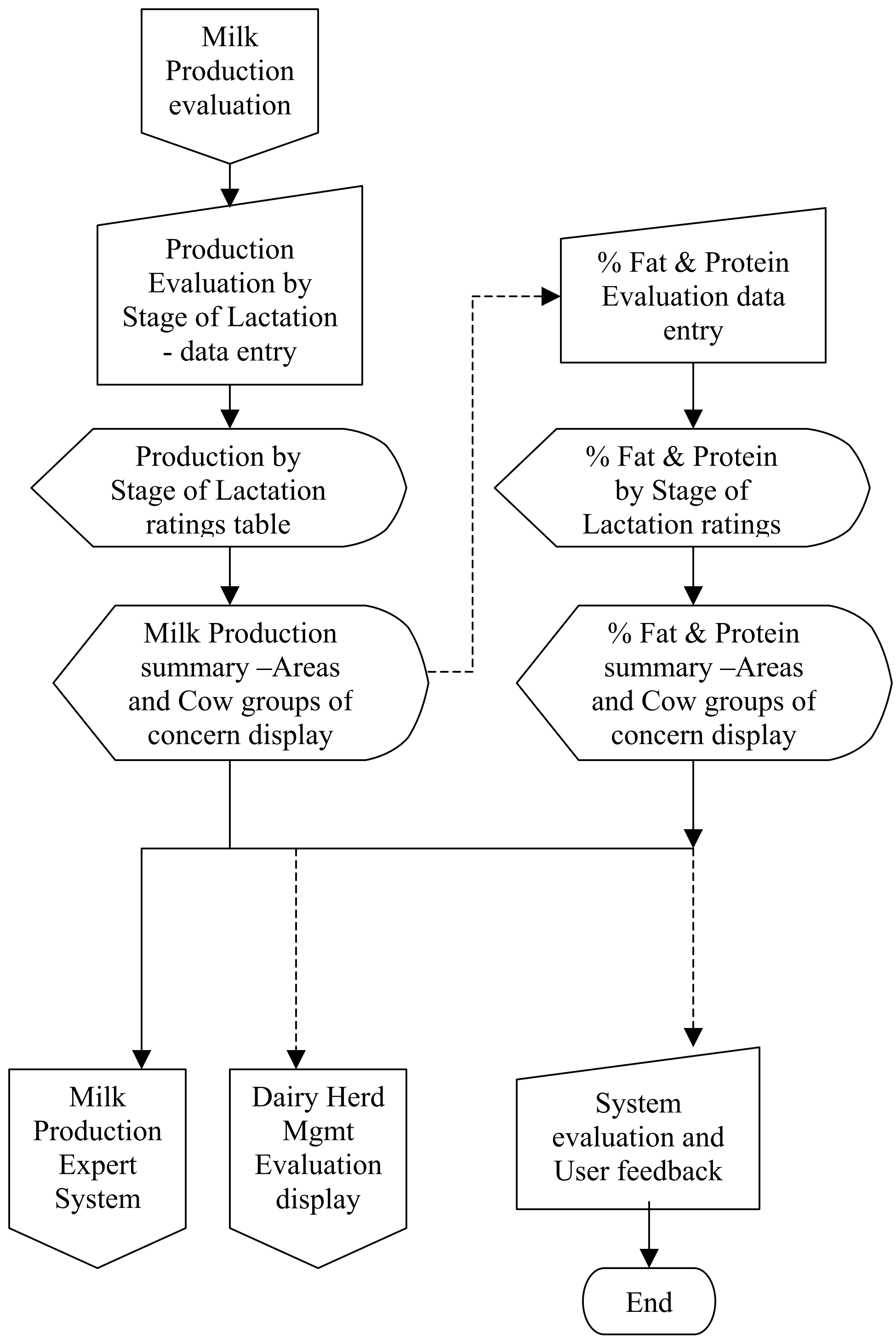


Figure 2.3: Detailed evaluation: Milk Production analysis

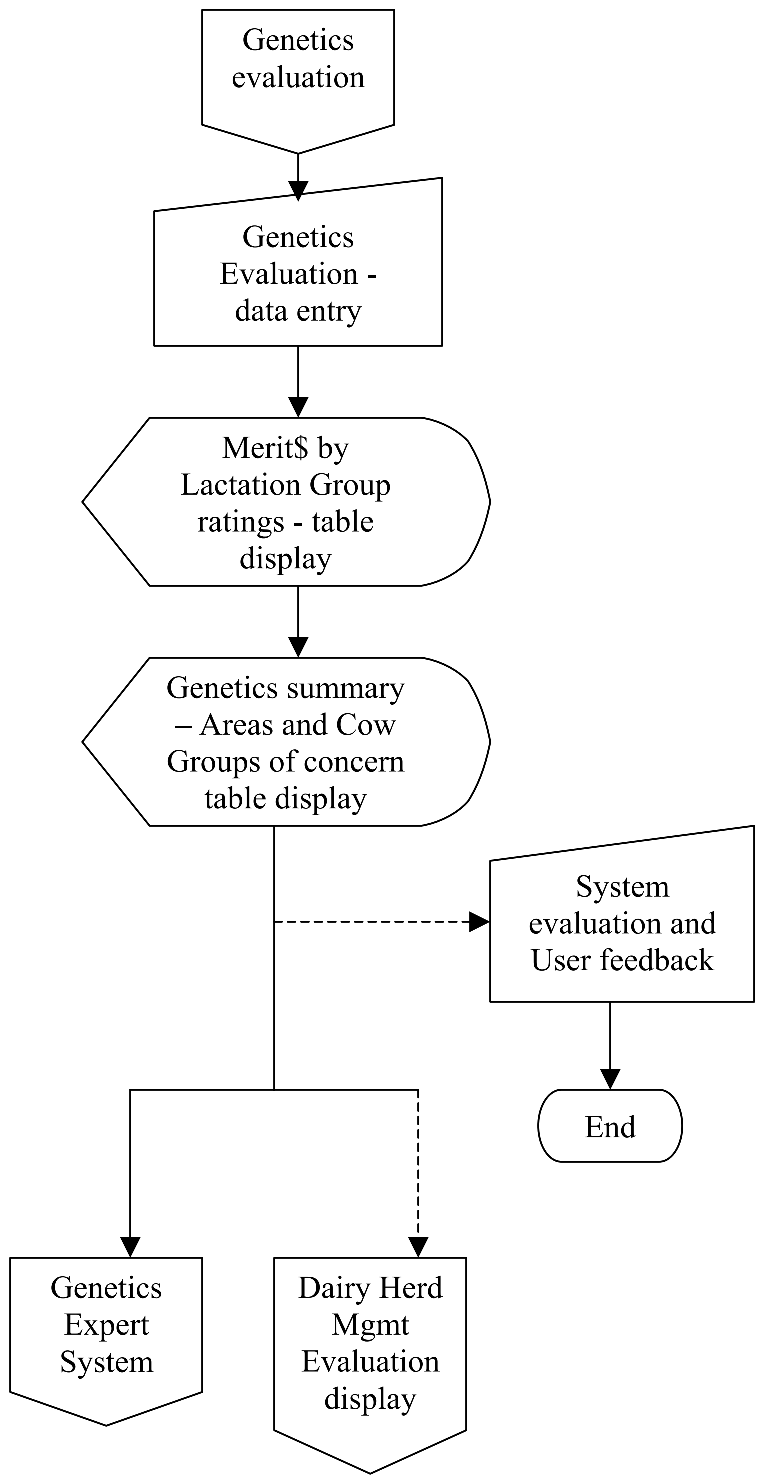


Figure 2.4: Detailed evaluation: Genetics analysis

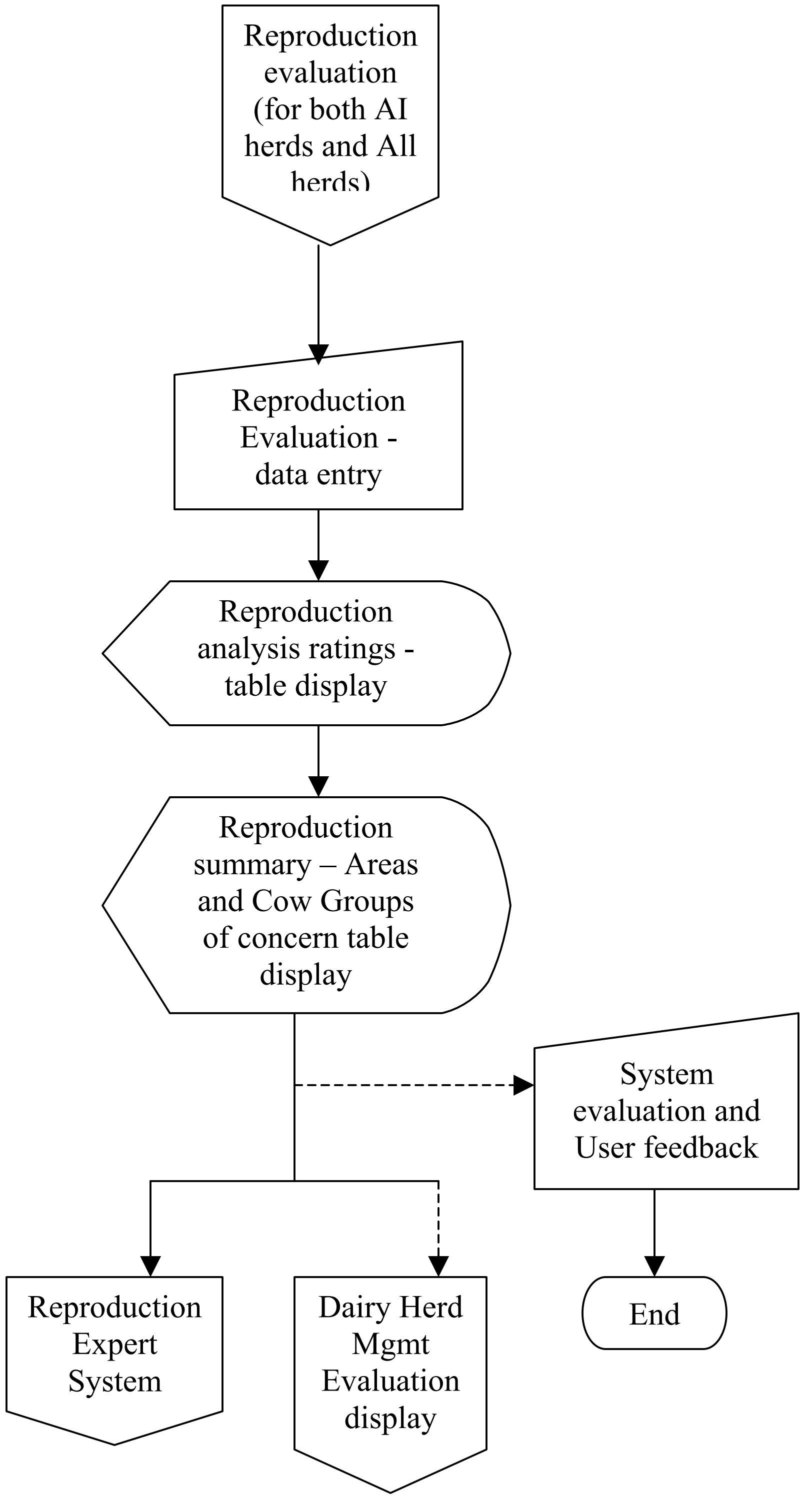


Figure 2.5: Detailed evaluation: Reproduction analysis

2.2 Implementation

The preliminary benchmarking analysis has been implemented using Web-programming technologies. Simple static web content (cf. the Welcome page, Expert System entry pages, system evaluation and user feedback) has been coded using the HyperText Markup Language [HTML401 1998], Cascading Stylesheets [CSS2 1998] and the industry standard Javascript [ECMA262 1999]. Sun Microsystems' Java servlet technologies [Servlet23 2002] have been followed in generating on-the-fly dynamic content for the rest of the analyses, since it is different for each producer. An open-source, free, robust implementation of the Java Servlet specification from the Apache Software Foundation, Tomcat [Tomcat33 2002] was used as the servlet container which executes the servlets and seamlessly passes the output to the HTTP server, the Apache httpd program [httpd 2002]. The Apache httpd server and the Tomcat servlet engine have been so designed as to work together – the HTTP server is configured to serve static content, and seamlessly pass on servlet requests to Tomcat which also performs load-balancing to reduce the load on a server. The DairyMAP system operates on this recommended Apache httpd-Tomcat configuration, on a Microsoft Windows 2000™ host. Servlets allow for dynamic on-the-fly generation of content, which is required in this case since the system responds to each user on an individual case-by-case basis. In early versions of the system, proprietary versions of a Java servlet engine and webserver were tried, which prompted the need for a system based on more secure, robust, transparent implementations of Java technologies and the webserver. Another feature in the design of DairyMAP is session expiry, which discontinues access after a period of inactivity by the user during a session. The user is then required to re-authenticate in order to use the system in such instances. These features of user registration, login, session expiration, and user feedback were implemented by Yuki Ono. Some screenshots of the system in operation are shown next.

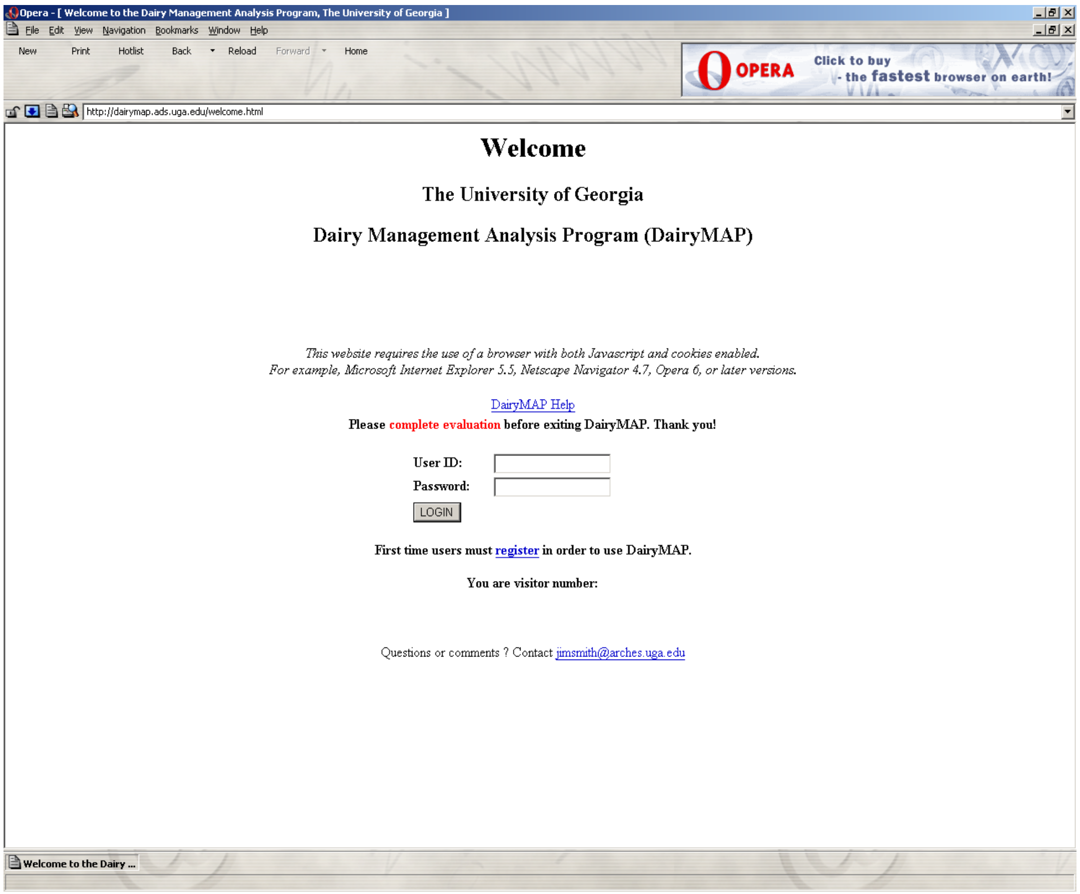


Figure 2.6: DairyMAP: Welcome page and login screen

This figure shows the entry page of the system where a registered user can login to use the system. A hyperlink explaining what DairyMAP is about takes the user to a help page which contains background information and further, to a sample DHI-202 report. A producer can locate (looking at the sample) the values to be input in his/her own copy of the report. A user who has not yet registered can also do so (at no cost) from the link provided to the registration page. Figures 2.7 and 2.8 show some typical examples of herd information that a producer may input into the system.

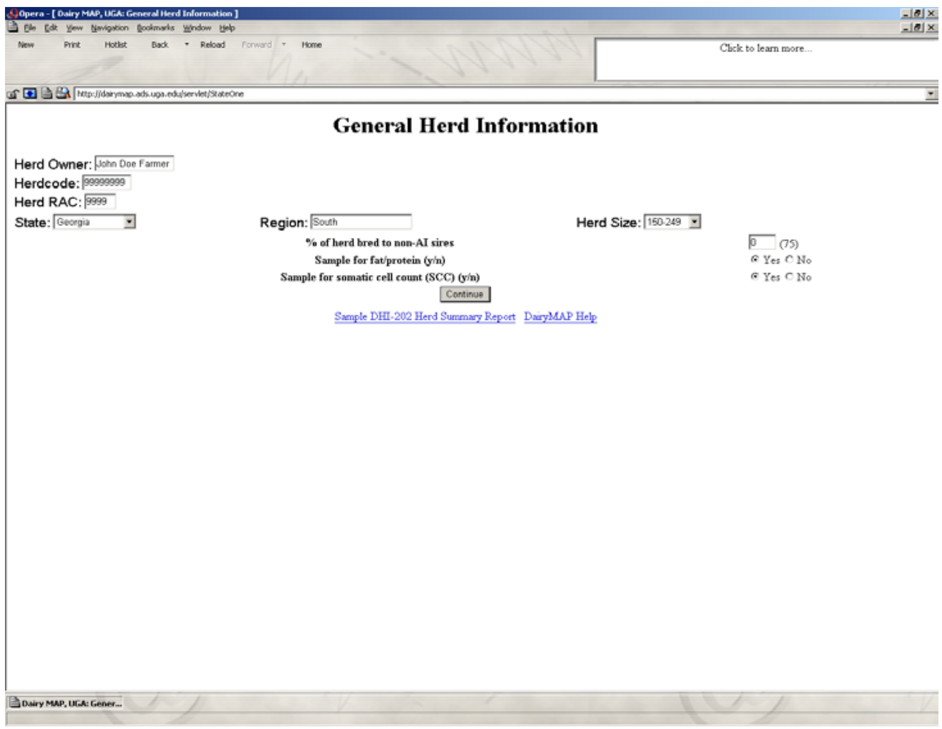


Figure 2.7: General Herd Information data entry

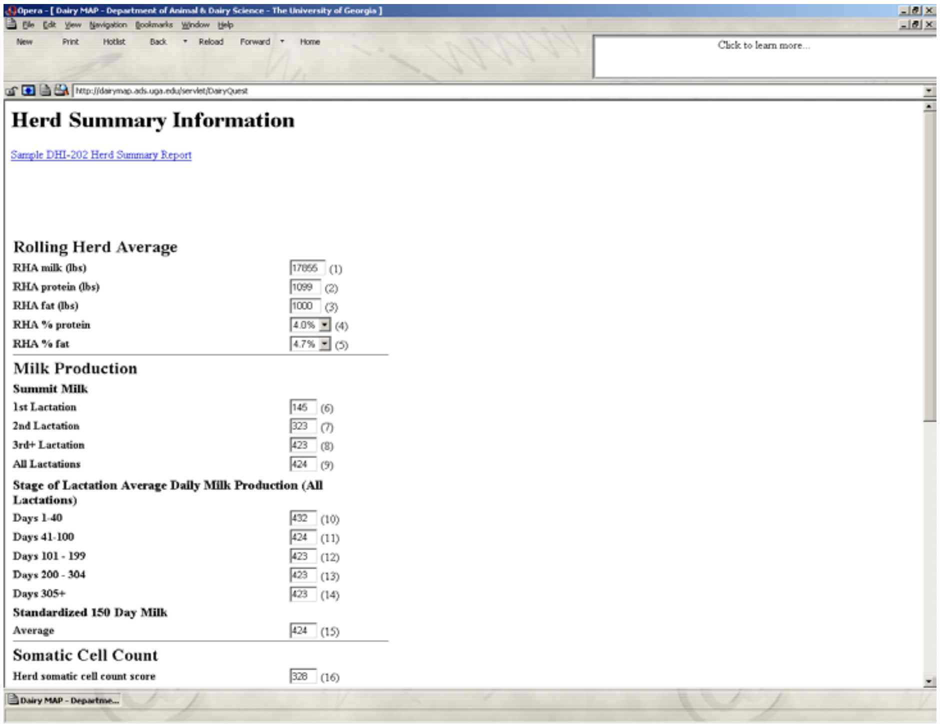
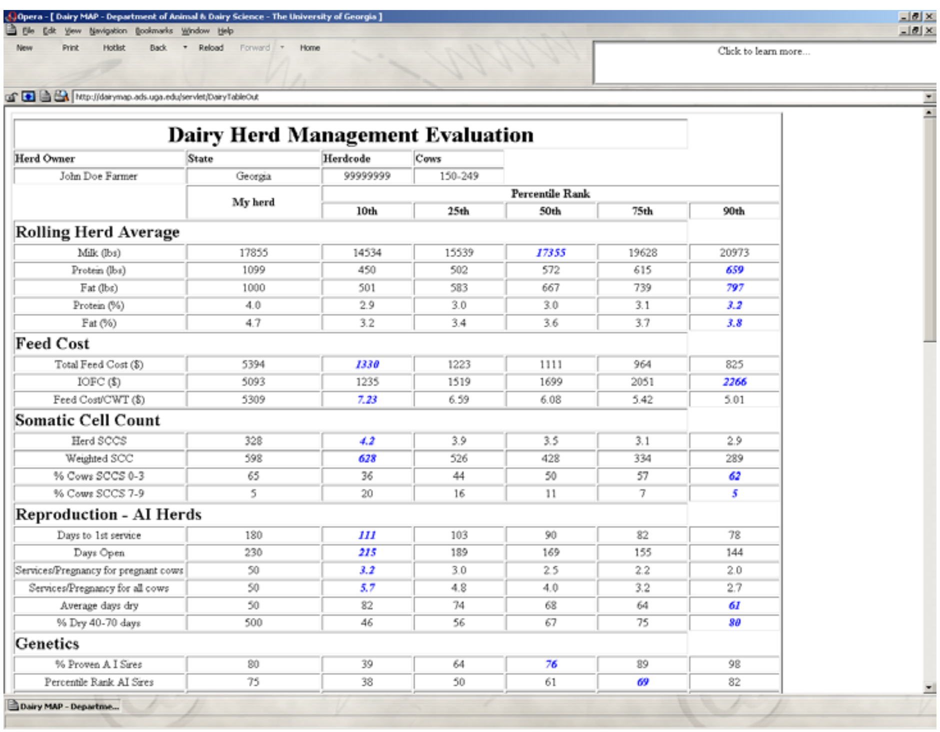


Figure 2.8: Herd Summary Information data entry



Herd Owner	State	Herdcode	Cows
John Doe Farmer	Georgia	99999999	150-249

	My herd	Percentile Rank				
		10th	25th	50th	75th	90th
Rolling Herd Average						
Milk (lbs)	17855	14534	15539	*17355*	19628	20973
Protein (lbs)	1099	450	502	572	615	*659*
Fat (lbs)	1000	501	583	667	739	*797*
Protein (%)	4.0	2.9	3.0	3.0	3.1	*3.2*
Fat (%)	4.7	3.2	3.4	3.6	3.7	*3.8*
Feed Cost						
Total Feed Cost ($)	5394	*1330*	1223	1111	964	825
IOFC ($)	5093	1235	1519	1699	2051	*2266*
Feed Cost/CWT ($)	5309	*7.23*	6.59	6.08	5.42	5.01
Somatic Cell Count						
Herd SCCS	328	*4.2*	3.9	3.5	3.1	2.9
Weighted SCC	598	*628*	526	428	334	289
% Cows SCCS 0-3	65	36	44	50	57	*62*
% Cows SCCS 7-9	5	20	16	11	7	*5*
Reproduction - AI Herds						
Days to 1st service	180	*111*	103	90	82	78
Days Open	230	*215*	189	169	155	144
Services/Pregnancy for pregnant cows	50	*3.2*	3.0	2.5	2.2	2.0
Services/Pregnancy for all cows	50	*5.7*	4.8	4.0	3.2	2.7
Average days dry	50	82	74	68	64	*61*
% Dry 40-70 days	500	46	56	67	75	*80*
Genetics						
% Proven A I Sires	80	39	64	*76*	89	98
Percentile Rank AI Sires	75	38	50	61	*69*	82

Figure 2.9a: Herd Management Evaluation display

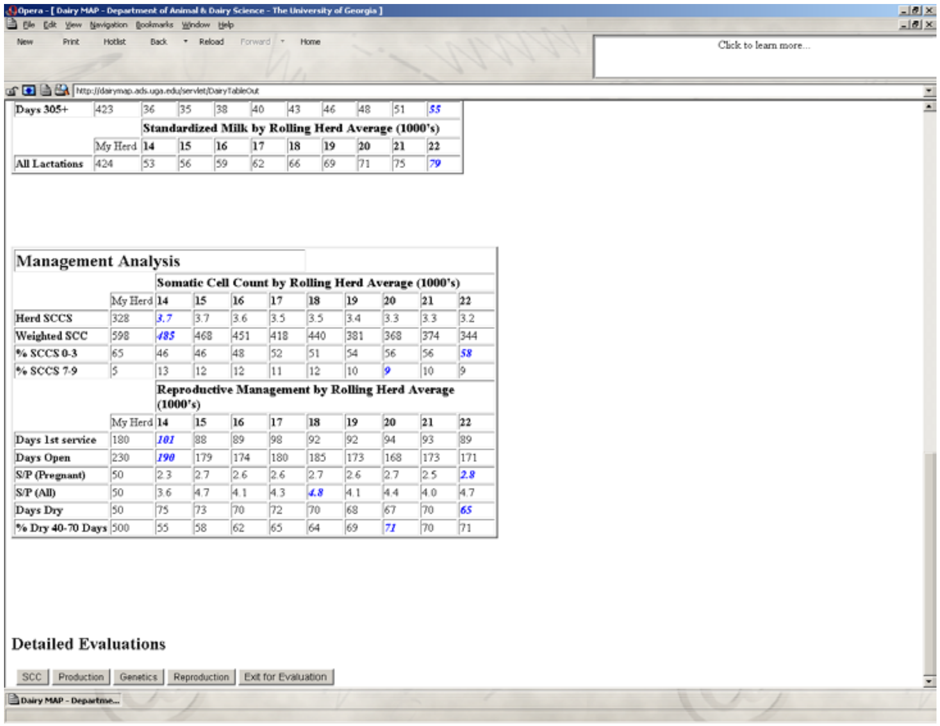


Figure 2.9b: Herd Management Evaluation display

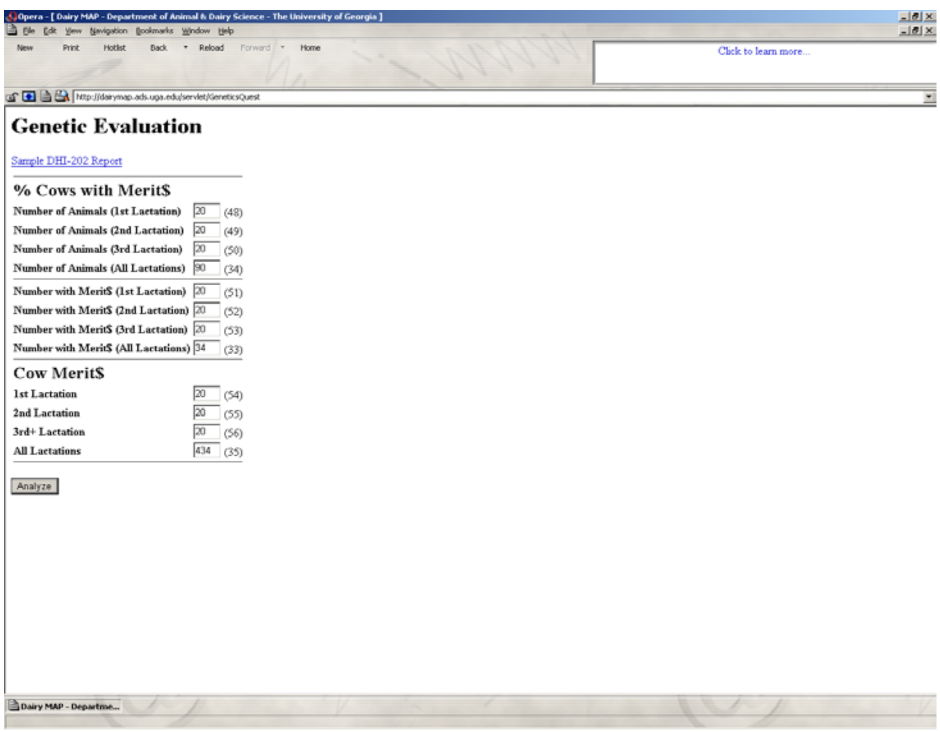


Figure 2.10: Detailed Evaluation: Genetics data entry

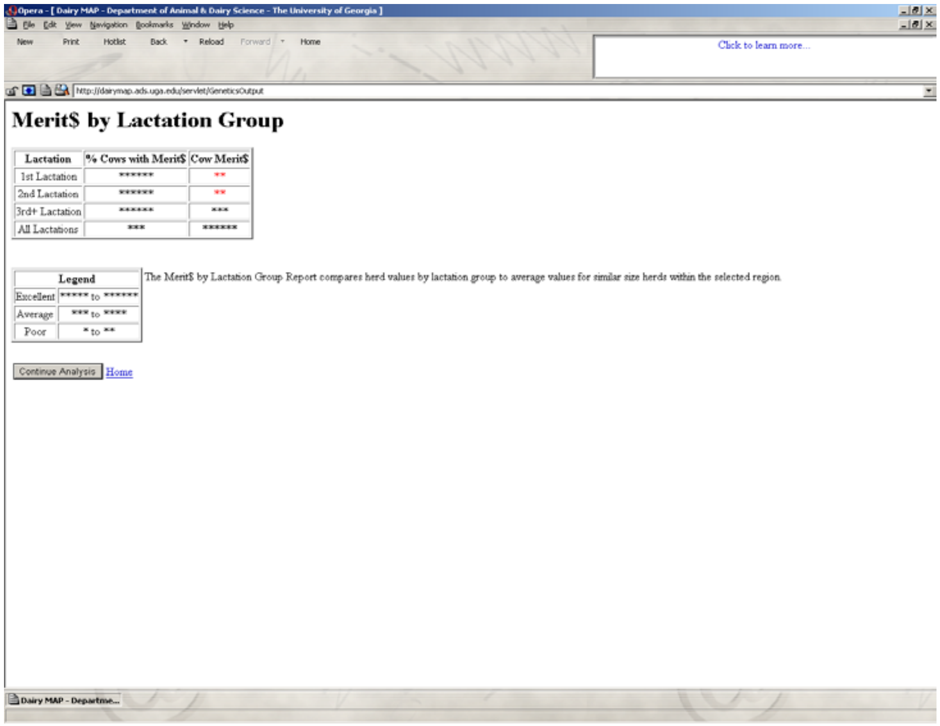


Figure 2.11: Detailed Evaluation: Genetics – Merit$ by Lactation Group display

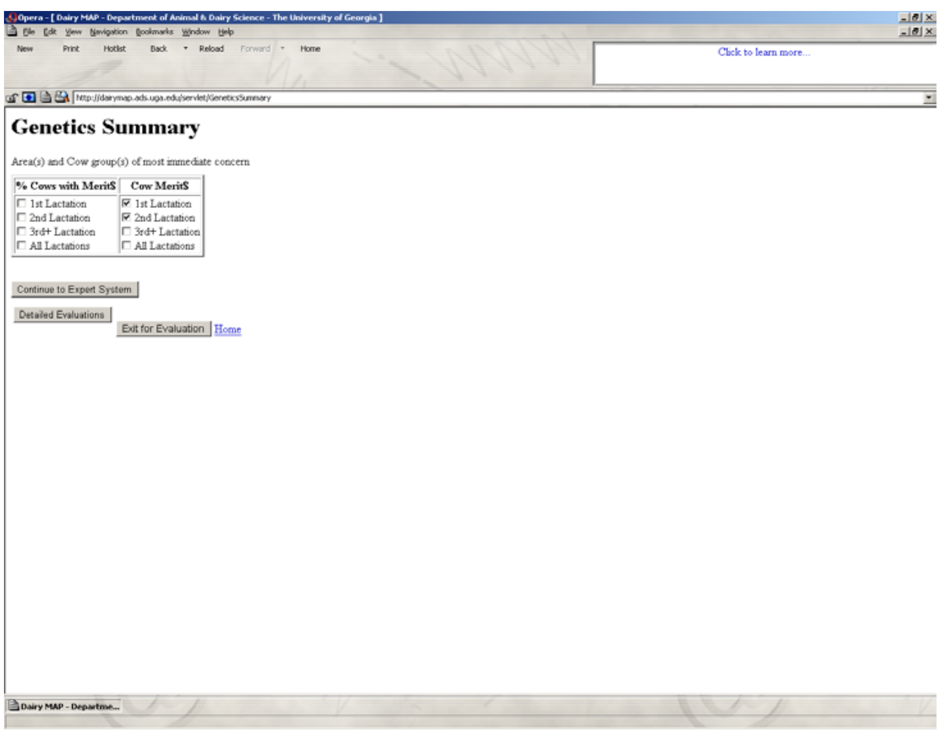


Figure 2.12: Detailed Evaluation: Genetics – Summary display

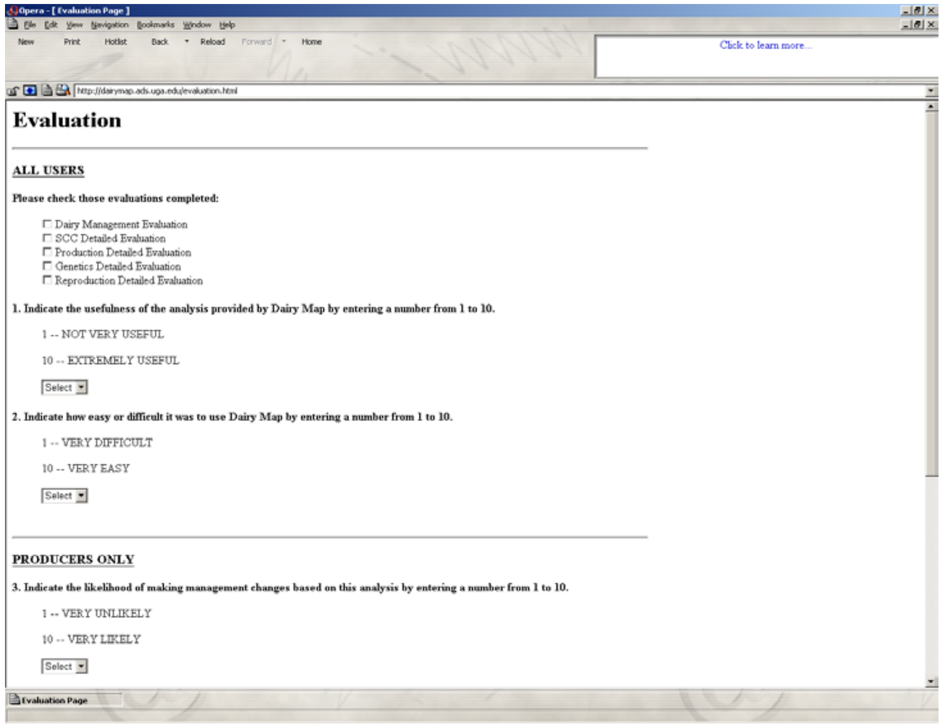


Figure 2.13: User feedback and evaluation

Figures 2.9a and 2.9b show an intermediate herd management evaluation stage, actually displayed on the same page but split into two screens for illustration purposes. The display shows those numbers (extracted from the dairy records database) italicized and blue-colored which are closest to the values for each of the parameters that the producer entered for his/her own herd. The producer can thus quickly identify the standing of the herd and the percentile it belongs to, in relation to similar sized herds in the region. Figure 2.9b shows a continuation of the herd evaluation table, along with the buttons that a producer can click on to proceed to detailed evaluations in each of the four major areas. The producer also has a choice to exit the evaluation completely and proceed to a user-feedback page by clicking on the "Exit for evaluation" button.

Figures 2.10, 2.11 and 2.12 show a further detailed evaluation in genetic characteristics of the herd, with typical values typed in. Figure 2.11 shows the "star" rating of the herd in this context, with areas of concern displayed in red color. Figure 2.12 shows the summary page with areas of concern checked. The producer can then either choose to go on to the expert system, on to detailed evaluations in the other major areas or to the user-feedback page to exit the session. Figure 2.13 shows a simple questionnaire and user feedback form for comments on the system which are e-mailed to the maintainer.

CHAPTER 3

THE EXPERT SYSTEM: DESIGN AND IMPLEMENTATION

3.1 Overview

The Expert System starts where the Web-based preliminary analysis ends, using the summary page of areas of concern in the preliminary analysis to determine the specific questions to ask of the dairy producer. Since this is a Web application, an Expert System environment which is capable of communicating via the HTTP protocol the World Wide Web is based on, is needed. From prior tried-and-tested use of the Logic Programming Associates Ltd. (UK) LPA ProWeb Server™ (version 4.110) at the Artificial Intelligence Center for such applications, it became a natural choice for DairyMAP. ProWeb is a specialized extension to a Hypertext Transport Protocol (HTTP) server. ProWeb is written in the PROLOG language, and supports the development, testing and deployment of intelligent, dynamic server-based applications on intranets and the Internet. The LPA ProWeb Server [Proweb 2001] allows web sites to use the powerful reasoning capacity and backtracking capabilities of PROLOG. PROLOG formed a good choice because of its in-built back-tracking, pattern-matching, unification capabilities, along with an appropriate form of knowledge representation.

The DairyMAP system integrates the core expert system with the question asking process. The system creates each question for the user, and simultaneously processes the information it has gathered so far to generate intermediate conclusions. By the time the user finishes answering all the questions, the system has already analyzed a majority of the data and is ready to present some of its evaluations, by considering the answers to all the questions together. At the end of each module, the expert system provides a page summarizing the expert's recommendations, the reasons for those recommendations, and any further suggestions for improvement. A module may be further divided into sections which address each part of the module separately. Some details about each of the modules are given below.

- Somatic Cell Count (or Mastitis) module: regarding the infectious disease that cows may develop during calving, lactation or the dry period, which manifests as an increase in the milk Somatic Cell Count. Questions on udder health and calving (pre-partum cows, new-born, nursing and weaned calves, and breeding heifers), housing (whether cows are housed in a proper environment), proper milking procedures, dry cow antibiotic therapy to combat infection, and culturing the milk samples to check for the presence of the disease-causing bacteria.
- Milk Production module: regarding the milk yield of a group of cows classified according to days in milk and lactation number. Questions are asked about the body weight of the cows, nutrition and feeding of cows (including protein content, roughage-concentrate ratio, feeding technique and availability of water).
- Genetics module: regarding how cows and heifers are bred for good genetic characteristics, including selection of sires and whether they are bred using natural service and/or artificial insemination.
- Reproduction module: consists of sections on Voluntary Waiting Period (relating to the interval between calving and first breeding), Heat Detection Efficiency (detecting cows that are ready to breed), Semen Tank (storage conditions of bull

semen to prevent degradation in quality), Insemination techniques (the way cows are artificially inseminated with semen from the sires), Herd Health (general suggestions and improvements to overall herd health), and Bull Management (how bulls are managed for breeding).

Within each section, the system accumulates information for that portion separately from the others and processes that data first, and then proceeds to any other sections that must be asked. Recommendations for problems diagnosed are displayed at the end of each phase.

3.2 Using LPA Win-PROLOG™ and ProWeb Server

There are some peculiarities in using ProWeb (due to the way the PROLOG language is designed) for Web-based interaction. Values of variables during the execution of a PROLOG clause are not available when the clause finishes execution. This episodic nature of data in PROLOG causes significant problems in storing values of variables across multiple clauses during a session. However, PROLOG and ProWeb circumvent the problem by explicitly asserting information pertaining to questions answered, as PROLOG clauses, in its internal database. This ensures that certain clauses are not executed more than once when not necessary, and also that multiple copies of data are not stored. This ensures that there is no degradation of performance since the language is based on pattern-matching and unification, relying on searching for clauses asserted and stored in the internal database. The ProWeb Development Environment provides for such checks implicitly in the way HTML Forms are served to the user, through the use of ProWeb-specific predicates. For example, consider the following clause:

```
start :-  send( form1 ), form_returned( form1 ),
          send( form2 ).
```

If `form1` has not been sent yet, the clause `form_returned(form1)` fails and execution stops. Once it has been sent to the client and returned, the second clause succeeds and the rest of the `start` clause is executed. This functionality is used to control execution of the program. The clauses

```
proweb_returned_answer(<question>, <value>)
proweb_returned_form(<form_name>)
```

check the internal database to see if the answer to `<question>` or the form `<form_name>` has been sent and received yet. These clauses placed before certain points of the program help to ensure that data are not analyzed until it has been accumulated. An additional PROLOG construct, called `assert`, is used in many cases to control the flow of the program execution and store values to be used later by the expert system.

3.3 Testing and debugging the expert system during development

The ProWeb environment allows for testing the expert system during development without the need for an Internet connection. The necessary components for this are – the LPA Win-PROLOG Development Environment™ along with the ProWeb Server, a Web browser capable of supporting HTML Forms (*e.g.* Opera Software's Opera, Netscape Navigator, Microsoft Internet Explorer), and the expert system source code. A special ProWeb clause called `proweb_nn_register` (for the Opera and Navigator browsers) and `proweb_ie_register` (for Internet Explorer) connects the

Web browser to LPA PROLOG. It should be noted that the browser must be running *before* executing the clause in LPA PROLOG. From then on, answers to questions contained in the HTML pages would be communicated directly to LPA PROLOG, and one can test/debug the expert system code. At the end of a session, this connection needs to be explicitly broken by calling the corresponding `proweb_nn_unregister` or `proweb_ie_unregister` clause, *before* halting the Win-PROLOG environment and closing the LPA application.

3.4 How ProWeb generates web-pages

The ProWeb environment provides for a way to generate web-pages with HTML forms, and process the information returned to it by the client when a HTML Form on the web-page is submitted. Each HTML page generated by ProWeb consists of three components – header, body, and footer. HTML code at the very beginning of a page (part of the header), at the start of the body, and that at the end of a page remain constant for all pages, and hence these three small snippets of code are stored separately in three different files named appropriately. The clauses relevant to this level of page construction are the following:

```
proweb_page( [<form name>],  [page construction] )
proweb_form( <form name>,  <form structure> )
form_title_file( <form name>, <title> )
```

The first clause defines what pieces for the page are needed and how they are put together. The second argument of the clause is the layout of the web page generated by the ProWeb environment, and it is returned as a list of ordered elements that indicate which HTML tags are to be inserted in that order. These tags are then converted into proper HTML tags by ProWeb when the page is prepared to be sent to the browser. It is within this clause that the user defined clause form_title_file is called to extract the title of the page and the three snippets of HTML code above are included into the page. The clause proweb_form is used by proweb_page when it invokes proweb(<form name>). At this stage of creating the page, the variable <form name> is used to reference all the parts needed to put the page together. The value of <form name> is passed as a parameter directly from proweb_page to form_title_file and proweb(<form>). The names need to be consistent, otherwise, the attempt to instantiate the title or the form will fail, resulting in an error message to the user. The second argument of the second clause defines how the questions are displayed in the form. The general form for the output is:

```
[table, tr, td, <question text_1>, td, ?<question name_1>
      ...
          tr, td, <question text_n>, td, ?<question name_n>]
```

A question-asking page consists of a table of questions followed by a row of buttons to advance to the next page. Each question takes up a row, and the user is presented first with the actual text of the question stored in the variable <question text>, followed by the structure of the answer stored in the variable <question name> from which he/she can choose an answer. This stage of construction is done using the clauses makeQuestion and generateQuestionForm, as below:

```
   makeQuestion( <question name>, <question structure> )
generateQuestionForm( <Questions>, [<Question Form Structure>])
```

The question structure is split into several parts and the constituent pieces are referenced by <question name>, the value of which needs to be consistently identical, as with the case of <form name> mentioned previously.

The clause makeQuestion generates the question, implemented as a row in the table, by calling the clause questionText. The actual text presented to the user when a question is asked is selected from stored clauses of the form:

```
questionText( <question name>, <question text> )
```

where <question name> serves as a name of the question to ask. It then prefixes a question mark to the value of <question name> as required by ProWeb, and adds it to the list to be returned. The clause generateQuestionForm takes a list of questions returned by makeQuestion, adds the row of buttons generated by invoking the clause generateButton, and a table of previously answered questions before it, using the clause displayAnswers. The final form of the page output to the user is:

Answered Questions (via displayAnswers clause)

Table of New Questions

Row$_1$ (question$_1$) (via makeQuestion)

Row$_2$ (question$_2$) (via makeQuestion)

…..

Row of Buttons (via generateButton)

Finally, the ProWeb environment replaces the occurrence of ?<question name> with the answer type the user selects when answering the question. The format of the contents of the answer structure is determined by the clause proweb_question, wherein the value of <question name> is used to refer to the correct answer type for the question.

```
proweb_question( <question name>, <question structure> )
```

The type of answer may vary (*e.g.* a value that the user may type in, or, a choice that can be made from a pull-down selection menu). For example, a menu-box type of answer filled with `[option1, option2]` with a preset selection `option1` is coded as:

```
[ method = menubox, select = [option1, option2], prefill = option1 ].
```

3.5 File Structure (processing the answers given by the user)

The structure of the files that perform the analysis for each set of questions in a module is essentially the same. The first portion defines how the execution flows from one stage to the next. After checking whether this stage has been done before, the system generates and stores recommendations about each specific criterion first, and then follows with an analysis of the questions for that particular section as a whole. Each module has been coded into a group of files which are named according to the content and function of the code in them. The main clause of each module loads the rest of the files for the module. The question-asking sequence, the actual text of the questions, the rules which decide which action to perform next (*e.g.* go on to the next question, terminate question-answer session and generate recommendations, and so on), and some lower-level clauses for debugging as well as error-checking, are all coded into separate files for good modular structure. The rationale for such structuring is to have a central location for certain clauses and corresponding values. For example, if the output needs to be displayed in a different format, only the clause relevant to creating the output (which is coded in a single location in a single file) needs to be changed.

Each successive step in the analysis involves extracting the answer to a question and then calling clauses with that answer as the argument. Each specific clause has one defined for a blank answer where no information is stored since that question was skipped as well as answers to questions for which the user did provide a response. Questions with non-blank answers generate recommendations and suggestions for improvement only when the value of the answer is not optimal for that criterion. If the answer is in fact the best possible, that fact is stored, and positive feedback is provided to the user commending the answers, and is used in the overall general section evaluation. However, if the user had not answered any questions, then nothing is displayed after the analysis.

The rest of this section briefly describes the files used in the analysis by the expert system, using the Reproduction module as an example. Files used to generate the question pages and the corresponding inference files are both described. For more detail, the files contain well-written comments. Each subsection of a file starts with a detailed summary of how the execution flows from one clause to the next, and is followed by how the clauses specific to that file are used.

3.5.1 Reproduction module: Main start clause (as related to inference)

File Name: `mainproc.pl`

Description:

The main start clause in this file checks to see which questions are to be asked. The expert system only makes analyses on information it has in its database, so this first step limits the amount of inference it does. The decision as to which questions will be asked is based on the results of the previous knowledge-based section coupled with the area(s) that the user chooses for further analysis. The present implementation is a page with drop-down boxes in which the user selects a 'yes' or 'no' in answer to whether to proceed further. The start clause calls another clause which asserts into the internal database the clauses `askwaiting`, `askheat`, `askBullMgmt`, `askhealth`, `asksemenTank`, `askartinsem`, with either a yes or no as their argument and then proceeds to the clauses to ask each section of questions. These correspond to questions about the sub-sections about Voluntary Waiting Period, Heat Detection Efficiency, Bull Management, Herd Health, Semen Tank, and Insemination techniques respectively. The Voluntary Waiting Period sub-section is briefly explained as an illustration of how each sub-section is organized.

3.5.2 Reproduction module: Voluntary Waiting Period sub-section

Questions File: `waitingQuest.pl`

Description:

The first part of this file determines which questions regarding Voluntary Waiting Period conditions, if any, are to be asked. This stage is skipped if either the value of `askwaiting` in the internal database (determined by the main clause in `mainproc.pl`) is `no`, or the clause `waiting(done)` is defined in the database. Once the forms that contain the questions have been sent and received, the clause `waiting(done)` is asserted into the database to prevent a repeat of this stage when the main clause is invoked again.

The section dealing with form formats in this file define the way that each of the forms in the question-asking sequence (`waiting_form1`, `waiting_form2`, and so on) is created. The content of some of the pages in the sequence varies depending on answers given by the user in previous pages. The last part of this file contains clauses that define the options available for the user to select when answering questions. For example, the clause `daysChoices` contains the choices for the number of days a producer is willing to wait to breed a cow after she has calved.

Inference and Analysis File: `waitingRules.pl`

Description:

This file deals with the conclusions drawn from analyzing information gleaned from the file `waitingQuest.pl`. The rules are invoked only if there is a definite `yes` or `no` answer, or a definite numerical answer where required, to the question. Two rules presented here for illustration are:

1. Voluntary Waiting Period (VWP):

Condition	Conclusion	Recommendation	Reason
VWP greater than 65 days.	VWP conditions are poor.	Consider reducing the waiting period to less than 60 days and begin breeding healthy animals sooner.	Improving heat detection and Artificial Insemination techniques may help.
VWP is between 50 and 64 days.	VWP conditions are poor.	You may want to consider breeding some animals before 50 days.	Improving heat detection and Artificial Insemination techniques may help.
VWP is less than 49 days.	VWP conditions are ideal.	None.	VWP in use is the ideal, recommended value.

2. Days to first service: Days to first service for current herd = A, Yearly average days to first service for total herd = B

Condition	Conclusion	Recommendation	Reason
(A = B) and (A >= 65)	Days to first service conditions are poor.	Consider breeding sooner; lowering voluntary waiting period and improve heat detection and fertility.	None.
(A – B) >= 5	Days to first service conditions are poor.	Emphasize breeding cows earlier since Days to 1st service has increased significantly.	Improving management techniques to breed animals sooner
A >= 85 or B >= 85	Days to first service conditions are poor.	Consider breeding animals sooner after calving.	Possible problems with heat detection and Artificial Insemination techniques.
A >= 145 or B >= 145	Days to first service conditions are poor.	Consider lowering voluntary waiting period and improve heat detection.	Improving management techniques to breed animals sooner.

The rules and files relating to the other sub-sections (Heat Detection Efficiency, Bull Management, Herd Health, Semen Tank, and Insemination techniques) are similarly organized.

3.6 Constructing the output recommendations as a web-page

The file which consolidates all the information entered by the user, constructs the web page and presents the results as a web-page served by the ProWeb Server is `output.pl`. This file contains the clauses used to present the conclusions at the end of an expert system session. The clause `proweb_form` is used to specify how to construct the output form, and the page is divided into parts corresponding to the sub-sections used in the module. For example, the Reproduction module consists of sub-sections Voluntary Waiting Period conditions, Heat Detection Efficiency, Bull Management, Herd Health, Semen Tank, and Insemination techniques. The process is as follows:

1) Obtain recommendations for each sub-section in a general sense. If any of the specific criteria have a recommendation (and also evaluated as poor), then the clauses return the message stating that the user can improve in this area.

2) Retrieve the list of recommendations filtered with the general sub-section in mind.

3) Create the page with general recommendations for each section followed by specific information for that section.

If the producer is doing well for a criterion, he gets a message indicating so, but if he does not answer any questions, then there is no output from the expert system. This latter aspect needs to be changed in the future, perhaps, to a warning message stating that the effectiveness of the expert system is reduced by not answering any questions.

3.7 Linking the Expert System with the Web-based preliminary analysis

The results of the preliminary benchmarking analysis are intended to guide the user to a specific area of questions the expert system would ask and analyze the answers. Section 3.5 described how the system determines which areas have been selected and how it starts to ask the questions relevant to that area, with the Reproduction module as an example.

The original idea for the link was to simply indicate which areas the system suggests are in need of further study and the producer can deselect those options and/or select ones that were not included. For example, suppose the initial screening process establishes that the herd SCCS (Somatic Cell Count Score) is high for first lactation cows in early lactation. The producer is presented a list consisting of checkboxes for 1) calving condition, 2) housing, 3) milking procedure, and 4) dry cow treatment in which the first option is selected (and the rest are not). The assumption here is that the expert system will ask questions only about calving conditions, since the source of the infection is most likely from poor facilities for cows while they are calving. If the herd SCCS is high for those cows in later lactation, the reason could be the result of exposure to more sources of the infection, mainly from milking procedures. More items in the list will be selected to indicate that possibility. The producer can then start the expert system portion of the web site with the options already selected, or he can deselect list item one and instead choose another item before proceeding. Currently, the system has start-pages for each of the expert system modules for each of the four areas, with options to select (or de-select) for individual conditions within the modules.

The following pages show some screenshots of the expert system Reproduction module in operation. Figure 3.1 shows the module start page that contains the questions for the sections of the module (*viz.,* voluntary waiting period, heat detection, artificial insemination, semen tank, bull management and herd health program). The dairy producer can choose the sections to ask questions on, by selecting ‘yes’ or ‘no’ from the drop-down selection boxes. The producer can also choose to answer questions from all the sections in this module by selecting ‘yes’ to the last question. Figure 3.2 shows a page during the question-answer process in which the producer has answered some of the questions, with further questions to be answered. Most of the questions can be answered simply by choosing the choices provided in the drop-down selection boxes. The most common choices have been coded into the drop-down selections, while text-input boxes allow the producer to type in numbers for questions which require numerical input. In these intermediate question-answer stages (Figure 3.2), the system accumulates answers to questions already asked, and forms partial conclusions based on the answers thus far. The questions in the sequence also change depending on the answers provided. For example, if the producer does not have a herd health program (where herd health is supposed to be monitored regularly), then the system would not ask further questions as to whether pregnancy checks are conducted or how often the herd is checked.

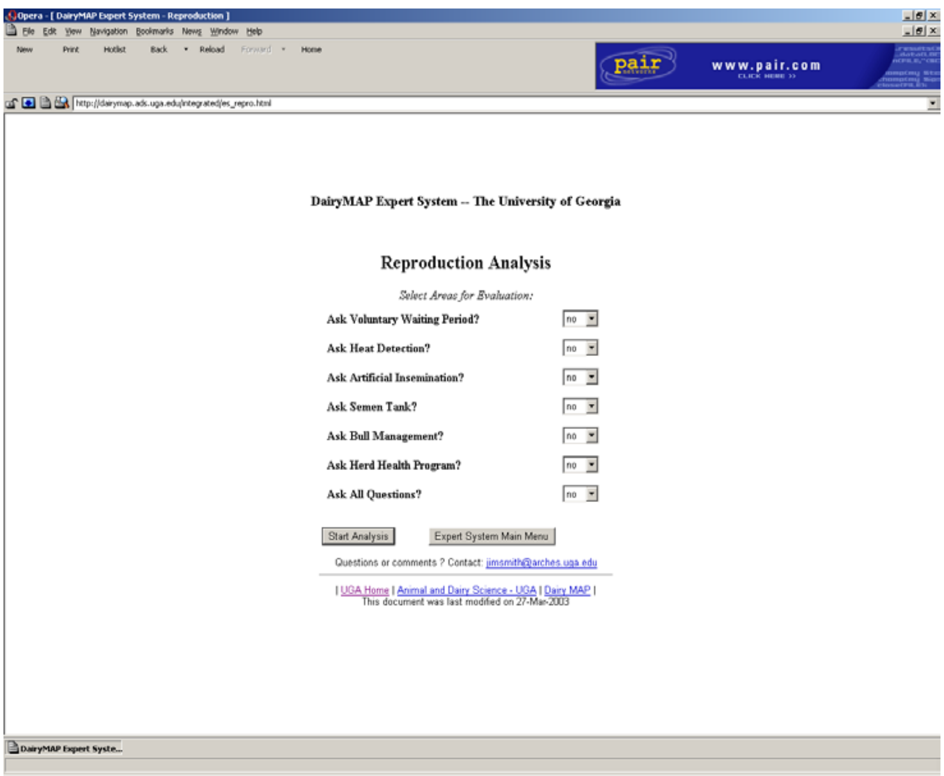


Figure 3.1: Expert System Reproduction module start page

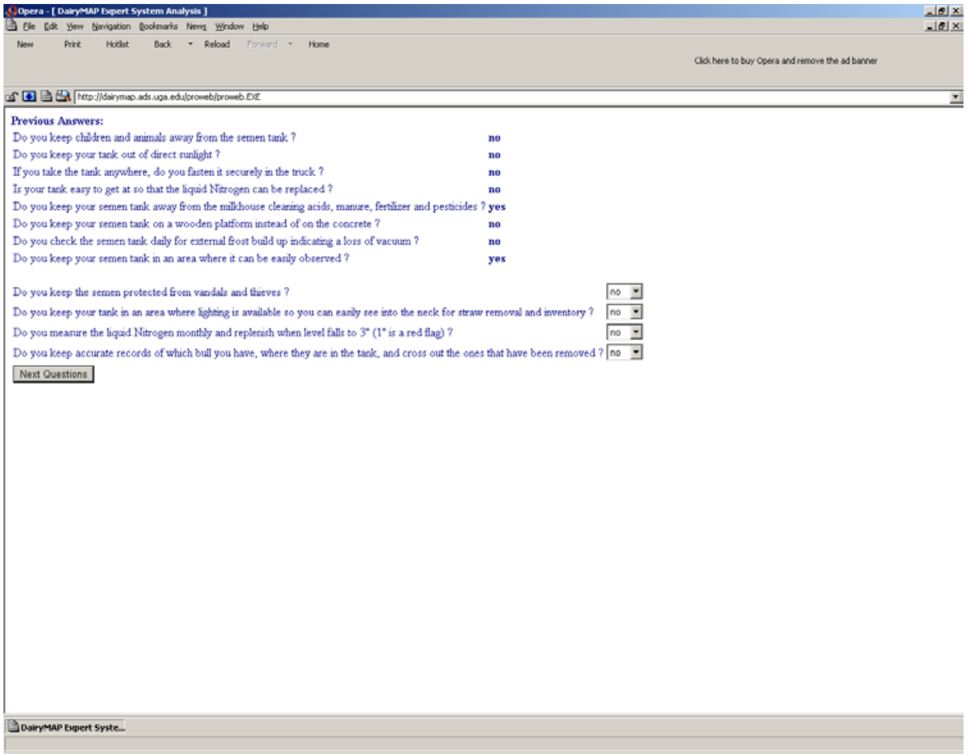


Figure 3.2: Expert System Reproduction module question-answer sequence

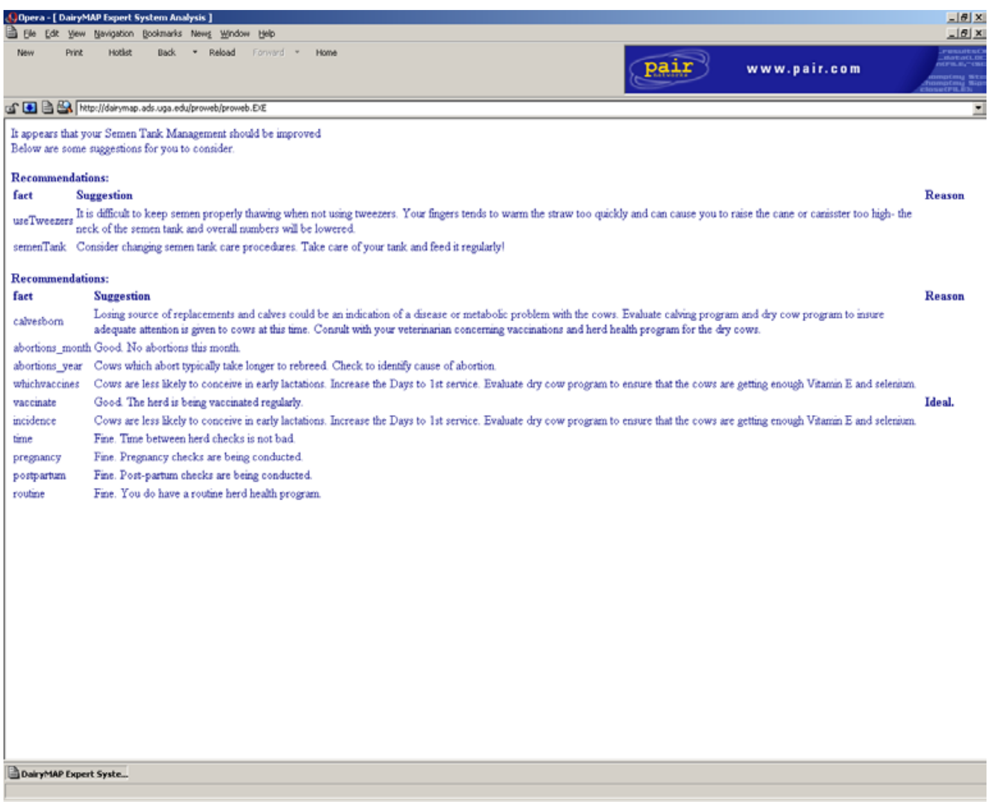


Figure 3.3: Expert System Reproduction module recommendations

Figure 3.3 shows a screenshot of the final recommendations page at the end of a module, where some comments on the condition of the producer's herd in the particular context are made. These are displayed in a tabular format, with the PROLOG fact/clause coded in the system corresponding to the question/answer, the expert recommendation/suggestion, and a reason (if any). Thus each question/answer would correspond to a row in the table. If more than one sub-section in a module has been analyzed, then each set of recommendations is displayed in tabular form one after the other, as shown above. In this figure, the set of recommendations about the semen-tank section are displayed on top, and the recommendations about herd health displayed below. The producer can then return to the expert system start-page from where he/she can proceed to other modules of the expert system. Currently, the expert system start page is not hyperlinked to the page where the final recommendations are displayed. This may be addressed in future work.

CHAPTER 4

CONCLUSIONS

The goal of this work was to design, develop, implement and test a WWW-accessible, benchmarking analysis and rule-based Expert System to help dairy producers in improving the quality of their herds. The system uses Web technologies to communicate with a user, who can be anywhere in the country. The system consists broadly of two major components - the benchmarking statistical analysis and the Expert System. The preliminary analysis is carried out by comparing a producer's herd data from a Dairy Herd Management Report (also called a DHI-202 report) with herds of similar size in the same region, and shows specific areas of concern that a producer needs to focus on. The system then allows the producer to proceed to the expert system which diagnoses herd problems in four major areas of dairy management, and displays recommendations. Though there have been similar efforts in dairy management by others, focusing in individual areas, this is a single, integrated management system providing guidance in all four major aspects of dairy management *viz.,* Milk production, Somatic cell count / Mastitis, Genetics, and Reproduction.

The system has been implemented with Java servlets on the Apache web-server (version 1.3.27) and the Tomcat servlet container (version 3.3) for the benchmarking analysis component, and PROLOG code on the Logic Programming Associates Ltd. (UK) LPA ProWeb Server™ (version 4.110) for the Expert System. The Apache/Tomcat web-server/servlet-engine combination is a recommended, robust, scalable, open-source configuration, allowing for load-balancing, developed and released by the Apache Software Foundation. They are so designed that Apache and Tomcat can each be running on different physical hosts, so that the load of the system can be distributed over multiple machines. The current system has Apache, Tomcat and ProWeb, all running on the same physical host. The system adheres to modular design, with logically-named files for easy maintenance, modification and addition of more modules to the benchmarking analysis as well as the Expert System. Most of the components of the system are based on open-source, free, robust implementations of the underlying technologies. This reduction on proprietary components confers reduced costs in implementing the system as an additional benefit.

The system currently has about 175 registered users (from the United States, Canada, Mexico, South America and Europe) who have provided some feedback. This work has thus been a successful demonstration of automating the expert diagnostic process within the domain of dairy herd management and can be treated as the first-step in a project that can be evolved further in expanding its capabilities. This work has consolidated all the four important aspects of dairy management (*viz.,* Milk production, Mastitis / Somatic Cell count, Reproduction, Genetics) and provides a single, integrated system for diagnosing typical management problems and suggesting remedial measures in helping dairy producers increase yields of their herds.

The system in its current implementation is easy for a programmer to maintain, modify the rulebase, and make changes as necessary. However, further work in the future may focus on a mechanism for the dairy experts to be able to modify the rule-set, and add questions and recommendations, preferably via a Graphical User Interface (GUI). The system can also be made to add and incorporate new rules, questions and recommendations from the users themselves, as another way of adding knowledge to the system. Such user contributions would, however, need to be validated by the domain experts. A possible approach to this would mean that an expert system shell with a knowledge-base (modifiable using a rule-editor) be used. Another possible deployment may consist of Apache web-server and ProWeb server on one physical host, with the Tomcat servlet engine along with the Dairy Records database on a second machine, to handle higher loads since database accesses consume significant system resources.

REFERENCES

[Allore et al. 1992] Allore, H. G., Jones, L. R., Merrell, W. G., Oltenacu, P. A., and Blake R. W., 1992. MAST, a data driven expert system to evaluate mastitis, *J. Dairy Science* 75: (suppl. 1): 268 (abstr.).

[CSS2 1998] CSS level 2 specification, 1998. W3C Recommendation, available at http://www.w3.org/TR/REC-CSS2/

[Demecq et al. 1991] Demecq, J. J., Nebel, R. L., McGilliard, M. L., and Pasquino, A. T., 1991. Expert system for evaluation of reproductive performance. *J. Dairy Science* 74: 3446.

[Devir et al. 1993] Devir, S., Renkema, J. A., Huerne, R. B. M., Ipena, A. H., 1993. A new dairy control and management system in automated milking farm: Basic concepts and components. *J. Dairy Science* 76: 3607.

[DHI202] DHI-202 Herd Summary Sheet, Dairy Records Management Systems Inc., Raleigh, North Carolina.

[ECMA262 1999] ECMA-262 JavaScript Language Specification, 1999, available at http://www.ecma-international.org/publications/standards/ECMA-262.HTM

[Feigenbaum, McCorduck and Nii 1988] Feigenbaum, E., McCorduck, P., and Nii, H. P., *The Rise of the Expert Company: How Visionary Companies are using Artificial Intelligence to achieve higher productivity and profits*, New York: Times Books, 1988.

[Fermanian et al. 1985] Fermanian, T. W., Michalski, R. S., and Katz, B., 1985. An expert system to assist turfgrass managers in weed identification, *Proc. 1985 Summer Computer Conference, Am. Soc. Agric. Eng.* (499).

[Fourdraine et al. 1990] Fourdraine, R. H., Tomaszewski, M. A., Taylor, J. F., Waiter, J. P., and Chavey E. G., 1990. Using a lactation curve analysis program to increase producer management awareness, *J. Dairy Science* 73: (suppl. 1): 184 (abstr.).

[Helms et al. 1990] Helms, G. L., Richardson, J. W., Cochran, M. J., and Rester, M. E., 1990. Farm level expert simulation system to aid farmers selecting among crop insurance strategies, *Comput. Electron. Agric.* 4:169.

[Heueriser et al. 1993] Heueriser, W., Oltenacu, P. A., Guard, C. L., Holtz, C. R., Foote, R. H., Galton, D. M. and Mohammed H. O., 1993. A user-friendly computer program addressing the use of prostaglandin in dairy herd management, *J. Dairy Science* 76: (suppl. 1): 285 (abstr.).

[HTML401 1998] HTML 4.01 Reference specification, 1998.W3C Recommendation, available at http://www.w3.org/TR/html401/

[httpd 2002] The Apache HTTP server project, The Apache Software Foundation, 2002, available at http://httpd.apache.org/

[Kerr et al. 1999] Kerr, D. V., chaseling, J., Chopping, G. D., Cowan, R. T., 1999. DAIRYPRO – a knowledge-based decision support system for strategic planning on sub-tropical dairy farms. *Agricultural Systems* 59:257.

[Lemmon 1986] Lemmon H. E., 1986. COMAX: an expert system for cotton crop management, *Science* 233: 29.

[Levins and Varner 1987] Levins, R. A. and Varner M. A., 1987. An expert diagnostic aid for reproduction problems in dairy cattle, *Conput. Electron. Agric.* 2:47.

[McKinion and Lemmon 1985a] McKinion, J. M., and Lemmon, H. E., 1985. Expert systems for Agriculture. *Conput. Electron. Agric.* 1:31.

[McKinion and Lemmon 1985b] McKinion, J. M., and Lemmon, H. E., 1985. Symbolic computers and AI tools for a cotton expert system, *ASAE Tech Paper* 85-5520. *Am. Soc. Agric. Eng.*

[Michalski et al. 1983] Michalski, R. S., Davis, J. H., Bisht, V. S., Sinclair, J. B., 1983. A computer-based advisory system for diagnosing soybean diseases in Illinois, *Plant Dis*., 4:549.

[Oltjen et al. 1987] Oltjen, J. W., Burdett, L. G. and Selk, G. E., 1987. Expert systems for culling management of beef cows, *Oklahoma Agric. Ext. Sta. Rep. MP 119.*

[Parmar et al. 1996] Parmar, R. S., McClendon, R. W., Potter, W. D., 1996. Farm machinery selection using simulation and genetic algorithms, *Trans. ASAE,* 39:1905.

[Pellerin et al. 1994] Pellerin, D., Levallois, R., St-Laurent, G., Perrier, J.-P., 1994. LAIT-XPERT VACHES: An expert system for dairy herd management, *J. Dairy Science* 77: 2308.

[Proweb 2001] WIN-PROLOG Proweb Server 4.1 user guide, Logic Programming Associates Ltd., UK, 2001.

[Schmisseur and Pankratz 1989] Schmisseur, E., and Pankratz, J., 1989. XLAYER: An expert system providing management advice to poultry layer managers, *South J. Agric. Econ.* (Dec)*:*183.

[Schmisseur and Gamroth 1993] Schmisseur, E., and Gamroth, M. J., 1993. DXMAS: An expert system providing management advice to dairy operators, *J. Dairy Science* 76: 2039.

[Servlet23 2002] Java Servlet technology, Servlet 2.3 specification, 2002. Sun Microsystems Inc., available at http://java.sun.com/products/servlet/

[Smith 1992] Smith, J.W., 1992. Using DHIA herd summary information to evaluate management practices, *J. Dairy Science* 75: 155 (abstr.).

[Smith 1993] Smith, J.W., 1993. Using graphs and summaries to evaluate herd management practices, *J. Dairy Science* 76:233 (abstr.).

[Smith et al. 1995] Smith, J.W., Ely, L. O., Gilson, W. D., Guthrie, L. D., and West, J. W., 1995. Dairy Map – Dairy Management Analysis Program, *J. Dairy Science* 78:284 (abstr.).

[Smith and Rodekohr 1996] Smith, J.W., 1992. DAIRYMAP, A Cooperative program brings individual attention to dairymen, *J. Extension* 34(4).

[Tomcat33 2002] Apache Tomcat version 3.3, The Jakarta project, The Apache Software Foundation, 2002, available at http://jakarta.apache.org/tomcat/