THE DEVELOPMENT OF AN AUTOMATED RESIDENTIAL EXPERT SYSTEM (A.R.E.S.)

A Senior Thesis

By

Pablo White

1997-98 University Undergraduate Research Fellow

Texas A&M University

Group: Engineering I

The Development of an Automated Residential Expert System (A.R.E.S.)

by

Pablo E. White

Submitted to the
Office of Honors Programs and Academic Scholarships
at Texas A&M University in College Station
in partial fulfillment of the requirements for the

1997-98 UNIVERSITY UNDERGRADUATE RESEARCH FELLOWS PROGRAM

April 20, 1998

Approved to as style and content by:

Dr. John Yen, Associate Professor
Department of Computer Science

Susanna Finnell, Executive Director
Honors Programs and Academic Scholarships

Engineering I

Acknowledgments

I (Pablo White) would like to recognize the following people for their gracious contributions to this project:

Dr. John Yen, Associate Professor of Computer Science and
Director of the Center for Fuzzy Logic, Robotics, and Intelligent Systems;

Ron Sasse, Director of the Department of Residence Life and Housing;

Fred A. Wells, Computer Services Manager of the
Department of Residence Life and Housing; and

Cecelia Hawkins, Senior Lecturer of English.

Table of Contents

Table of Figures

Abstract

Since the pioneering endeavors of Warren McCullock and Walter Pitts (involving computer learning) in 1943, artificial intelligence has fascinated experts and has aroused the interest of the general public. Currently, computer science researchers are employing this vibrant technology to the following areas: expert systems, fuzzy logic, neural networks, robotics, vision, natural language, speech recognition, and genetic algorithms. As a society, artificial intelligence has prompted intense philosophical discussions involving logic versus emotion, computer predictability versus human spontaneity, and the characteristics of true intelligence.

Our project involves expert systems. Expert systems owe their popularity to the following features: their capacity to "mass produce" knowledge, their reduced cost of information decimation, their ability to work in environments deemed as hazardous to people, their permanence of expertise, their increased reliability (especially when humans would fail due to stress or fatigue), and their capability to explicitly explain every logical step from hypothesis to conclusion. At present, research efforts are working to resolve some of the disadvantages involving expert systems--mostly dealing with a computer's inherent inability to adapt to new situations and to detect incomplete or erroneous data.

The overall goal of our research project is to develop A.R.E.S., an expert system that will solve a problem within on-campus residence hall affairs.

Chapter 1: Introduction to Expert Systems

We begin our report with a brief overview of expert systems, a branch of artificial intelligence. An expert system solves problems by applying a series of rules to a given set of facts to produce diagnoses.

An expert system consists of seven basic components: an inference engine, an agenda, a knowledge base, working memory, an explanation facility, a knowledge acquisition facility, and a user interface (refer to Figure 1).

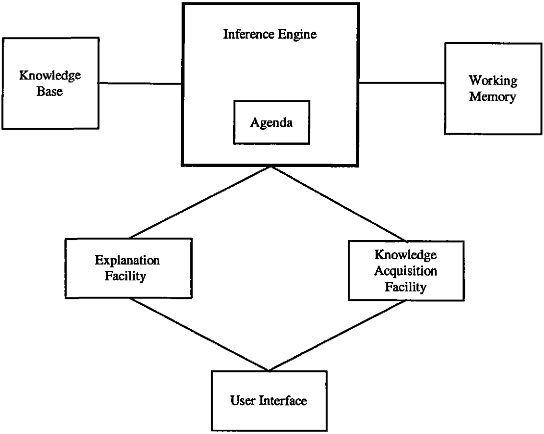

Figure 1: Components of an Expert System (Adapted from *Expert systems: Principles and Programming*)

The inference engine, the most important component, handles the expert system's analysis by deciding which rules are satisfied by facts, prioritizing the satisfied rules, and executing rules with the highest priority. The prioritized list of rules, stored in the knowledge base, comprise the agenda. The expert system uses the working memory to store facts and logs its reasoning steps in the explanation facility. The knowledge acquisition facility supplies an automatic way for the user to enter knowledge into the system. Finally, the user interface manages communication between the user and the expert system.

At the core of an expert system is various rules and facts. A rule simply consists of an antecedent (also called a conditional part, pattern part, or left-hand-side) which outlines necessary criteria and a consequent (also called the actions or the right-hand-side) which lists desired actions to be performed. For instance, a rule to guide an "intelligent" automobile driver through a controlled street intersection might be "IF the light is red THEN stop the car." In this case, "the light is red" is the antecedent, and "stop the car" is the consequent. A fact describes the state of a situation and can be a constant, a variable, or an object. For example, "the light is red" is a fact. Such a fact would case the "intelligent" automobile driver to apply the brakes and halt all forward motion of the car.

To apply rules to facts to arrive at a diagnosis, an expert system uses one of two methods of reasoning: forward chaining and backward chaining. In forward chaining, the expert system scans the knowledge base for satisfied rules and progressively executes the satisfied rules until it produces a final conclusion. During backward chaining, the expert system begins with a desired resolution and determines the rules necessary to produce that desired resolution. Then, the expert system works backward, matching conditions to necessary rules, to find the corresponding set of facts. If the original, given facts matched those generated via backward chaining, then the desired resolution is possible.

At present, expert systems have found industrial applications. Notable intelligent systems include: MYCIN (used to detect and treat bacterial infections), DENDRAL (interprets molecular structure), CRYSALIS (interprets the three-dimensional structure of a protein), and MetaCrawler (a World Wide Web search engine located at "http://www.metacrawler.com/"). Other areas include the automotive industry, the manufacturing industry, the medical profession, and computer games.

Chapter 2: Purpose of A.R.E.S.

2.1 -- The Problem.

We--Dr. John Yen as Director of the Center for Fuzzy Logic, Robotics, and Intelligent Systems and I as a Resident Advisor employed by the Department of Residence Life and Housing (to be hereafter referred to as Residence Life) at Texas A&M University--have become interested in the current room assignment process used to fill vacancies within Texas A&M's on-campus residence halls (refer to Figure 2). The present process grants priority to roommate preferences and allows applicants to select the "potluck" option. If a person states a roommate preference, that person most likely receives his or her selection as his or her roommate. However, if an applicant chooses the "potluck" option, Residence Life staff assigns him or her a roommate at random.

Despite the efficiency and simplicity of this process, this system presents two unwanted side effects--the presence of a moderate number of roommate conflicts within the residence halls and an incredible amount of work for the Residence Life staff.

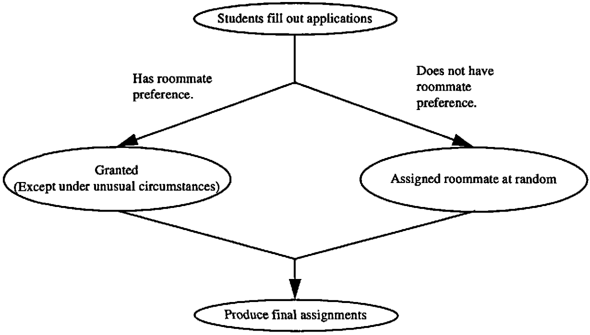

Figure 2: The current room assignment process used by Residence Life.

2.2 -- Our Proposed Solution.

Our proposed solution is to replace the "potluck" aspect of Residence Life's current room assignment system with A.R.E.S., an intelligent agent that will couple compatible persons into residence hall rooms (refer to Figure 3). We believe that this will introduce a pro-active step by reducing the number of roommate conflicts within the residence halls and will lessen the amount of work demanded by the Residence Life staff.

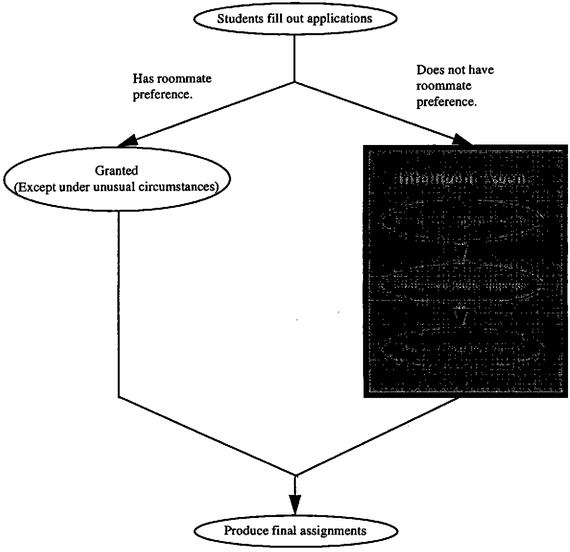

Figure 3: The proposed room assignment process for Residence Life.

From our conversations with Residence Life personnel, we have determined that A.R.E.S. must meet certain requirements for its adoption. First and foremost, its user interface must be intuitive and easy to use. Also, A.R.E.S. must be able to interface with computer systems currently in use by Residence Life (which possess

Pentium processors running Windows 95 or NT). In addition, the prospective resident selection process must not utilize any forbidden type of discrimination (e.g., race, creed, sexual orientation, etc.) and must be adaptable to meet Residence Life's current and future needs. Finally, due to immense amount of applications received by Residence Life each year, A.R.E.S. must possess strong scalability and must produce its recommendations in a reasonable amount of time--not days or months.

CHAPTER 3: DESIGN OF A.R.E.S.

3.1 -- High-Level Design.

We commenced our development efforts by formulating a high-level design for A.R.E.S (refer to Figure 4). First, we determined that the student data, the facts, would be stored in the working memory and would be sorted by a preprocessing algorithm. Then, the sorted facts would be passed into A.R.E.S.' inference engine along with the matching criteria, the rules in the knowledge base. Next, the inference engine would apply the matching criteria to the student data to produce a list of recommended matches. Also, the inference engine would log its analytical reasoning steps into an explanation facility for the user. Finally, the user could either accept A.R.E.S.' recommendations or request another iteration from A.R.E.S. with the same data.

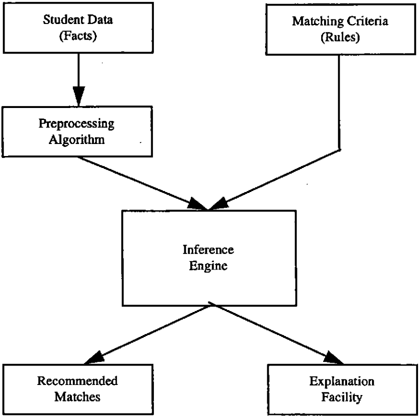

Figure 4: High-Level Diagram of A.R.E.S.

To promote intuitive and efficient communication between A.R.E.S. and its user, we outlined a graphical user interface (refer to Figure 5). A.R.E.S.' user interface would allow the user to enter or modify the student data and the matching criteria. Through such an interface, the user could manage files and interact with the inference engine. In addition, the user could print A.R.E.S.' recommended matches and other data.

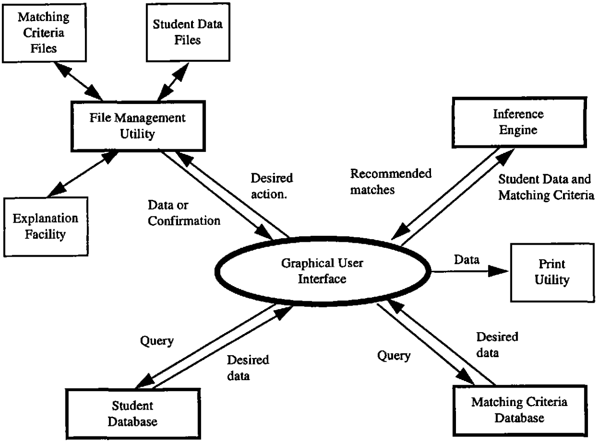

Figure 5: Graphical User Interface Interaction for A.R.E.S.

We intended to perform all initial development of A.R.E.S. on a Unix-based platform--due to its availability at the Center for Fuzzy Logic, Robotics, and Intelligent Systems. In particular, we determined to accomplish all programming in either C++ (for the user interface and the preprocessing algorithm) or CLIPS (for the inference engine), an expert system design tool developed at N.A.S.A. Eventually, we hoped to convert A.R.E.S. into an Intel-based Windows 95 or NT platform to ensure compatibility with the current computer system used by Residence Life.

3.2 -- The Preprocessing Algorithm.

Providing its recommendations in a reasonable amount of time became a problem for A.R.E.S. This problem mostly resulted from the large quantity of on-campus housing applications received by Texas A&M University each year. So, we investigated methods in which to preprocess the data for the inference engine.

Suppose that no preprocessing of data was to occur. A.R.E.S. would have to match prospective on-campus residents by "brute force," originating each match by scanning the data.

To illustrate this "brute force" method, let us consider an application pool with n applications. A.R.E.S. would have to compare one student with $n - 1$ other students to produce its first match. After forming its first match, A.R.E.S. could place aside those two students and continue with the remaining applicants. A.R.E.S.' subsequent matches would compare one student with $n - 3$ other students, the next student with $n - 5$ others, and so on until only two students are left (theoretically). A.R.E.S. would iterate $n / 2$ times to match every student and would perform

$$(n - 1) + (n - 3) + (n - 5) + \dots + (n - (n - 1))$$

comparisons to find matches. This results in a matching complexity of $O(n)$, linear with respect to the number of applications, for A.R.E.S.

From analyzing such a "brute force" method, we saw a direct linear relationship between the number of comparisons and the size of the application pool. We also recognized that no comparison method could perform superior to $O(n)$--since all of the applicants had to be viewed by A.R.E.S. at least once.

Due to the linear lower bound of A.R.E.S., we inserted a preprocessing algorithm to reduce the overhead of the initial matches. Our algorithm involved splitting the data into smaller categories. We had to ensure that our preprocessing would not invariably produce the matches--eliminating the usefulness of the inference engine. So, we ascertained that the applicants should be separated by sex and residence hall preferences, the discriminatory methodology already utilized by Residence Life.

3.3 -- The Inference Engine.

Once the preprocessing has been performed, the inference engine should formulate recommended matches. The rules should be the matching criteria, and the facts should be the student data.

To create its matches, the inference engine would match students by executing satisfied matching criteria. The inference engine would handle students by sex and hall preferences and would rank candidate matches by their fitness (or compatibility rating). Finally, the inference engine would formulate its final matches by placing as many students as possible in resident hall spaces.

Chapter 4: The Knowledge Engineering Process

The knowledge engineering process is a key step in the creation of an expert system. Knowledge engineering is the process by which the expertise from experts in the target field is quantified into a rule base for the inference engine. To assure the integrity of an expert system, the designer must ensure that any information obtained from the experts is definable via a series of rules or steps and is not incorrect or incomplete.

For our development of A.R.E.S., we chose to interview two experts: the Residence Life staff and the students of Texas A&M University. We believed that each of these groups presented us with a unique perspective regarding the room assignment process.

The Residence Life staff consisted of various members of the Residence Life computing staff and other personnel involved in the room assignment process. Even though all of those people surveyed readily noticed the need for A.R.E.S., they all presented us with divergent opinions regarding reforming their current system and the selection criteria appropriate for matching students. They expressed ideas such as using a student's academic classification, major, or other personal characteristics; but they possessed different priorities for their compatibility assessment criteria. However, they all agreed on the need for a student questionnaire (refer to Appendix A) to complement their room assignment process and enjoyed our suggested format for the questionnaires.

After speaking with the Residence Life staff, we wrote a survey consisting of multiple choice and short answer questions (refer to Appendix B). Our goal was to assess the ideal qualities students seek in their individual selection of roommates.

We presented our survey to a software engineering class at Texas A&M University. Our purpose was twofold: to obtain results and to test the validity of our survey. The students were cooperative, and we received 46 responses to our survey. Because of the diversity of responses (particularly to the short answer questions), we found it difficult to accurately tabulate the responses. However, we did discover that those students preferred roommates that were neat, held similar academic interests and lifestyles, and had previous positive associations with them. Furthermore, we ascertained that a large majority of those students desired a relationship of mutual respect and trust or a good friendship with their roommates (refer to Figures 6 through 8 for the survey results).

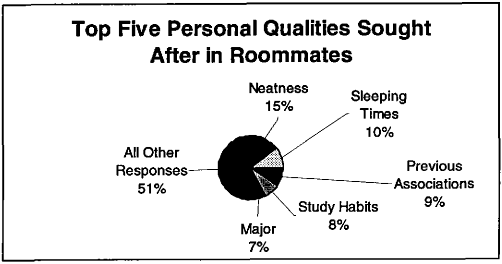

Figure 6: A breakdown of the responses to question 1 of the roommate preferences questionnaire (refer to Appendix B).

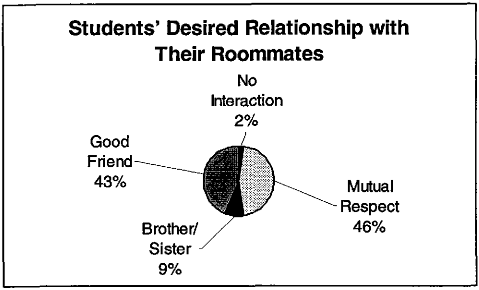

Figure 7: A breakdown of the responses to question 2 of the roommate preferences questionnaire (refer to Appendix B).

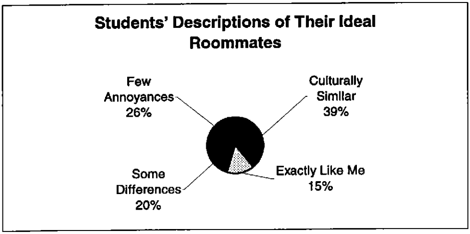

Figure 8: Breakdown of the responses to question 3 of the roommate preferences questionnaire (refer to Appendix B).

Chapter 5: Conclusion

As development of A.R.E.S. proceeded, we encountered challenges rooted in the subjective nature of our research project. The target requirements were difficult to quantify and prioritize, and the roommate preferences questionnaire absorbed a great amount of analytical effort. Also, the complexity of the knowledge engineering process hindered our progress. Thus, we were unable to produce a working prototype of A.R.E.S. because of the time constraints of this project.

We did, however, succeed in other facets of our research. We produced a viable design of A.R.E.S. with a sound algorithm. In addition, we formulated questionnaires and obtained results from students. Overall, we submitted a solution that would, in theory, solve Residence Life's room assignment system.

We hope that our research will serve as a foundation for future endeavors. We foresee many practical applications in public and private sector industries in addition to residence hall affairs.

References

Giarrantano, Joseph and Riley, Gary. *Expert systems: Principles and Programming.* Boston: PWS Publishing Company, 1994.

McLeod, Raymond Jr. *Mangement Information Systems.* Upper Saddle River, NJ: Prentice Hall, 1995.

Norvig, Peter and Russell, Stuart J. *Artificial Intelligence: A Modern Approach.* Upper Saddle River, NJ: Prentice Hall, 1995.

Appendix A: Sample Questions

Instead of responding to personal questions,
have the applicant respond to scenarios.

Why?

- Could avoid absolutes.
- Responses would be more accurate.
- Harder to "fix" results.

On weekends, I prefer to wake up …

A. Early
B. Mid-Morning.
C. Late
D. Sleep in until:

It's Saturday. You wake up and check your alarm clock. It's noon.

A. "I think I'll sleep a little longer."
B. "That's about my normal waking time."
C. "I missed the morning this time, but I'll Get up earlier next week."
D. "Oh my gosh! I've overslept."

How do you prefer your room?

A. Very clean.
B. Somewhat clean.
C. A little messy.
D. Very messy.

You walk into your room after attending class. Some books and papers are on the floor on your side of the room.

A. "… And your point is …"
B. "Oh well. I'll pick everything up later."
C. "I'll straighten everything up now before moving on to something else."
D. "That is NOT my room!!!"

Appendix B: Roommate Preferences Questionnaire

==

PROJECT A.R.E.S. (Automated Residential Expert System)
by Pablo E. White & Dr. John Yen

==

ROOMMATE PREFERENCES QUESTIONNAIRE

DIRECTIONS:

Please take a few moments to answer the following questions as honestly, clearly, and concisely as you can. By participating in this short survey, you will help us to quantify the criteria in which students select roommates. Thus, we truly appreciate your cooperation.

QUESTIONS:

1. Select the top five personal qualities (from these qualities listed below) that you utilize to select your roommate(s). (CIRCLE FIVE.)

-- Academic classification.
-- Amount of alcohol consumption.
-- Decorative tastes.
-- Extrovertive vs. Introvertive.
-- Hobbies.
-- Hometown.
-- Major.
-- Movie preferences.
-- Music preferences.
-- Percentage of daily hours spent in room.
-- Personal grooming.
-- Previous associations with person.
-- Relationship status (i.e., has girlfriend/boyfriend, does not have girlfriend/boyfriend, etc.).
-- Sleeping times.
-- Study habits.
-- Television entertainment preferences.
-- Other: (Please list.)

2. Which statement most closely defines your desired relationship with your roommates. (CIRCLE ONE.)
   A. "We do not interact at all. I go my way, and he/she goes his/her way. As long as he/she pays his/her fair share of living expenses, we are fine."
   B. "We both mutually respect each other. We sometimes interact and attend activities together."
   C. "He/She is like a brother/sister to me. We often confide in and seek advice from one another."
   D. "He/She is a good friend. We often eat dinner a lot and attend similar activities together."
   E. "We have significant differences that hinder total mutual respect. We rarely interact, but I at least know some detail about his/her personal life."

3. Which statement most closely defines your desired description of your ideal roommate. (CIRCLE ONE.)
   A. "I want someone who is culturally similar but at the same time possesses significantly different values than me."
   B. "I want someone just like me."
   C. "I want someone who has some significant differences from me."
   D. "I want someone who bears no similarity to me (different culture, language, etc.)."
   E. "I want someone who has a few minor annoyances."

4. Describe your best roommate.

5. Describe your worst roommate.

6. Define "cool".

7. Define "weird".