AN EXPERT SYSTEM FOR THE VALIDATION AND INTERPRETATION OF X-RAY RESIDUAL STRESS DATA

By

Marc J. M. Tricard

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE
in
Materials Engineering

APPROVED:

Robert W. Hendricks, Chairman

William T. Reynolds Donald R. Jensen

May, 1991
Blacksburg, VA

AN EXPERT SYSTEM FOR THE VALIDATION AND INTERPRETATION OF X-RAY RESIDUAL STRESS DATA

By

Marc Tricard

Committee Chairman: Robert W. Hendricks

Materials Engineering Department

(ABSTRACT)

Although widely recognized in the research community as one of the most accurate non-destructive methods for the determination of residual stress in polycrystalline structural materials, X-ray diffraction has not been widely adopted in the field. This is partly due to the fact that such measurements require, most often, a well trained user with knowledge in both materials and mechanical sciences in addition to the specific know-how of the instrument. We believe that computer assistance could contribute to the promotion of this technique by increasing the productivity and accuracy of these measurements. We have developed a prototype of an expert system, using Nexpert Object's shell, to assist a non-trained operator in the validation and interpretation of X-ray diffraction residual stress data.

The present work describes this prototype which has been designed to confirm the feasibility of the concept. Its knowledge base contains relevant examples of the rules necessary for data validation. The prototype has also validated most of the concepts required for the implementation of a full scaled version by evaluating all of the major technical features such as graphics representation, external routine calls and databases access. We have implemented significant rules to validate an

experiment, link our expert system with a database management system, develop a superset of data able to receive output from any existing X-ray machine and are working with a statistical pattern recognition software to discriminate between various d-vs-$\sin^2$Psi curves, to classify our data.

ACKNOWLEDGEMENTS

During this research program I have had the chance to work with a great advisor, Professor Robert W. Hendricks. I would like to thank him for everything he has done, from his administrative support to his infinite patience in explaining parts of his knowledge, even when I "asked too many questions!".

I also had two committee members Dr. Donald R. Jensen and Dr. William T. Reynolds who were willing to spend a lot of time helping me with my research. I have always been impressed by their ability to understand in 15 minutes what took me a month to grasp! I would like to express here my sincere gratitude for their support and guidance.

Thanks to my parents for their love, support and guidance.

I would also like to thank three colleagues, now my friends: Dr. Jinmyun Jo for his thoughtful answers to the thousands of questions I have asked him as well as for all the interesting open discussions we have had; Venkatramani S. Iyer for all his encouragement and his good humor; and to Hervé Mouille for his patience listening to me talk about my problems. Thanks as well to Ms. Theresa Johnson who helped me type the data into the database for the statistical analysis.

During the latter parts of this project I have been working with Scott Courtney and Jac Potet. They were working on the databases and files transfers while completing their Senior Project and Projet de Fin d'Etudes. I would like to thank them for their contribution and help with this research.

Thanks also to all the various experts worldwide who have answered our questionnaire: Mr. Michael Brauss, Dr. Jerome B. Cohen, Dr. V. Hauk, Dr. Michael R. James, Mrs. Beth Pardue, Dr. Paul Predecki, Mr. Paul Prevey, and Dr. B. Scholtes.

Thanks also go to Jeff Sanpore and Tom Morgan, both senior knowledge-engineering consultants, at Technology for Application Inc. for teaching me some of the mysteries of Nexpert Object...

My presence at VPI&SU and my participation in this project would not have been possible without the personal efforts of Dr. Robert Grynszpan. I would like to express to him my sincere gratitude. I still remember him telling me that it was possible to build an expert system on a subject without knowing anything about it. Thanks to Professor Garnier in ENSAM, Angers who taught me the basics of residual stresses and gave me much useful advices. Thanks also to Dr. Bernard La Berge for his administrative help, to Ms. Cindy Klein of the Materials Engineering Department for her eternal smile. Acknowledgement also goes to Ms. Jan Doran, Evelyn Janney, Suzette Weeks and to the ENSAM and VPI&SU administrations for their help.

I would also like to express my gratitude to the Société des Anciens Elèves des Arts et Métiers for their financial assistance, which has allowed me to "discover the new world".

Thanks also to Microsoft for the best debugging tool I have ever seen: Solitaire. Whenever everything goes wrong and you are thinking about going to Key West to practice boxing against a truck, a double click and hop Solitaire is inviting you to a small party to relax... Thanks to my computer for his cooperation, to Sub-Station for their famous "Today's special" and "Nachos", and to Backstreet Pizza for their twelve slices.

During this research I met many new friends: thanks to Angie, Antonio (the greatest debugger I know), Babu, Benoit (sa moutarde de Dijon, son "LAAAARD"), Christophe, Dilip ("The Beast"), Nikhil, Sandra, Steve (best hot dogs in town).

Special thanks to Hemanshu Bhatt, for everything...

I have taken most of the quotations figuring at the beginning of every chapter of this thesis from Raymond Kurzweil's book "The age of Intelligent Machines" (see [Kurzweill, 1990]. If you are only going to get a chance to read one book about Artificial Intelligence, this is the one.

This work has been made possible by the sponsorship of Caterpillar Tractor Company, Ford Motor Company, Rockwell International, and the Institute for Materials Science and Engineering of the Virginia's Center for Innovative Technology.

To Poochie ...

Table of Contents

1. INTRODUCTION

If machines could be so improved and multiplied, then all of our corporeal necessities could be entirely gratified, without the intervention of human labor, there will be nothing to hinder all mankind from becoming philosophers and poets.
Timothy Walker, essayist, 1831.

Machinery will perform all work - automata will direct all activities and the only tasks of the human race will be to make love, study and be happy.
The United States Review, 1853.

1.1 Presentation of the problem

The determination of residual stresses in crystalline materials, both metallic and ceramic, by x-ray diffraction is a well-established technique. Recent advances in both x-ray detector technology and in data acquisition hardware and data analysis software have resulted in instruments with capabilities only dreamed of just a few years ago. These advances now make it possible to measure in only a few seconds or minutes what used to take hours, thus making the large numbers of measurements required for on-line quality control or rapid component inspection possible. Notwithstanding these advances, there are also accompanying problems. Research over the past few years has shown that the simplifying assumptions of biaxial stress, small grain size, and isotropic materials generally incorporated into the underlying theory of the technique are often (and in some materials, generally) not met. Thus, data from real engineering samples and components are often complicated by such effects. This results in either misinterpretation by non-specialist operators in the field or in the need for having a local expert to assist in the analysis of the results.

There is a potential for the use of emerging artificial intelligence technology to develop an expert system to aid in the routine interpretation of such data. The potential for such a system, and a proposal for its implementation has been published elsewhere[1,2]. These documents provide an analysis of the user (customer) needs for such a system, the technology available to meet these needs, and the supporting justification for the conclusion that such a system is both needed and technically feasible. They then go on to provide the broad outline for the methodology of implementation of the proposed system and specific recommendations for its technical implementation.

1.2 Work schedule

The implementation of an expert system for stress determination using x-ray diffraction has been broken into multiple phases.

The first phase was the analysis of the problem and the identification of the potential solutions. This was

---

[1] See [Dehan, 1989-1].

[2] See [Dehan, 1989-2].

performed by Dehan in completion of his Master's thesis.

The second phase is the development of a prototype system using the components recommended in the previous work. This work is the culmination of the original project initiated by Hendricks and Roach under the sponsorship of Caterpillar Tractor Company, Ford Motor Company, Rockwell International, and the Institute for Materials Science and Engineering of Virginia's Center for Innovative Technology. The results of that development are the subject of the work presented here.

The third phase of the project will be to develop a full-scaled version of the expert system including all the features necessary for a complete analysis and interpretation of data. We will transform the first prototype into a field-operable expert system useful in the industrial workplace.

1.3 Artificial intelligence approach

The decision to use Artificial Intelligence ("AI") in this project has been taken in regard to the "human" interpretation and validation of data after any measurement of residual stresses by X-ray diffraction. For that reason all of the features of the prototype have been designed using only the AI software Nexpert Object[3]. But we will try not to forget Fox's law[4] :

"If an algorithm optimizes the solution, use it"

---

[3] For a quick introduction to Nexpert Object software see the "Introduction Manual" provided by Neuron Data.

[4] See [Fox, 1990].

Therefore, in the final version of our software we will try to restrain the AI part to only the real interpretation. Indeed all the statistical analysis and conventional (i.e procedural) programming, such as number crunching, will be performed with more efficiency with external routines. The concept is thus to develop a system where the AI part does not appear to the user but is embedded and only used - in a transparent way - as needed[5].

Indeed AI need not be seen as the perfect solution for the entire problem of data validation and interpretation, but rather as a powerful tool for only some aspects of it. An hybrid solution - conventional programming and AI - should lead to the best results.

1.4 Objectives of the project

This thesis corresponds to the Phase II previously mentioned. The goal of this work is to develop an operating prototype of the proposed expert system for the validation and interpretation of x-ray residual stress data. In such a development, it specifically has not been the purpose of this part of the project to develop any one of the features of the system in final form, but rather to develop an outline of a workable system with assurance that the

---

5 The author has seen for example the Audit Personnel Scheduler expert system developed by Technology Application Inc. This software assigns approximately 25 auditors of an internal audit department to the 100 to 150 audits that are scheduled throughout the year. The expert system assists in matching the needs of audits (i.e, foreign languages requirement, etc.) to the skills of individual audit teams. This system was implemented using Microsoft Windows, "C" language, and Nexpert Object. It has a spreadsheet-like user interface and the user is unaware that an AI core manages the system.

underlying concepts are sound and will meet the anticipated user needs.

The objectives of this prototype are thus:

. to implement a working prototype of the system which comprises a subset of the overall design,

. to write the system functional and detailed design specifications,

. to incorporate the C and assembly language routines programmed by Jac Potet[6] in our system. These routines import an experimental data file and transform it into the correct format (see further),

. to specify the structure of a Materials Database independent of any manufacturer which will be used as reference,

. to embed this Materials Database as well as an X-ray database, both developed by Scott Courtney[7], into our expert system,

. to verify and test the ability of Nexpert Object to handle several specific tasks such as database retrieval or graphics display, and

[6] See [Potet, 1991].

[7] See [Courtney, 1991].

. also major point to learn how to classify data either by statistical techniques, statistical pattern recognition, or AI/ES.

The next chapter presents, with a technical point of view, the prototype. Thus, we will show the hardware and software which has been chosen during this development stage, and describe the place of the prototype among an entire acquisition process. Then, after an explanation of our knowledge acquisition procedure, we will focus on Nexpert Object's knowledge representation with both rules and objects. We will also discuss outline the classification of data problem, and finally we will discuss the integration of all the components programmed into the shell.

2. SELECTION OF HARDWARE AND SOFTWARE

The world has changed less since Jesus Christ than it has in the last thirty years.
Charles Peguy, 1913.

Men have become the tools of their tools.
Thoreau.

In this chapter we describe the hardware and commercial software used during the development stage and further describe the relationships among them. Most of the software here described was suggested by Dehan in his thesis (see [Dehan, 1989]).

2.1 Hardware

The hardware selected for the project has been chosen to assure compatibility with the wide variety of commercial X-ray stress analysis instruments available internationally, to reduce cost, and to simplify implementation. It consists of:

. an IBM PC clone of the 20 MHz 386-class with 640 Kbytes of conventional memory and 5 Mbytes of extended memory, VGA color graphics, an 80 MByte hard disk, and a mouse,

. an HP LaserJet II printer, and

. to aid in the development and documentation of the system, an HP laser scanner with appropriate software for digitizing photographs and for optical character recognition (OCR) of printed information.

2.2 Commercial software

The commercial software selected for the prototype system includes:

. MS DOS 3.3 operating system with Microsoft Windows version 3.0,

. Nexpert Object[1] version 2.0 as the expert system shell under which the entire system operates,

. Microsoft Excel[2] version 3.0 as the spreadsheet,

. Micrografx[3] Designer version 3.0, to create graphics,

---

[1] Nexpert Object is an Expert System shell sold by Neuron Data, 444 High Street, Palo Alto, Ca 94301, Tel : (415) 321-4488, Fax : (415) 321-9648.

[2] Also sold by Microsoft.

[3] Micrografx, Inc, 1303 Arapaho, Richardson, Texas 75081, Tel: (214) 234-1769.

. Microsoft Paint[4] version 3.0 for the custom graphics displays,

. pLogic[5] version 1.1 as a statistical pattern recognition software, to help in the discrimination of data, and

. dBase III with Clipper 5.0 compiler, to create our database management system.

2.3 Hardware and software needed for the end-user

The previous list of hardware and software has been used to develop the prototype but it is expected that the final production version of our software will only require the following :

Hardware :

. An IBM clone with 4 Mbyte of memory in addition to the normal 640 Kbyte conventional memory, a 386 processor, a VGA display, a 40 Mbyte hard disk, and a mouse,

. a laser printer,

[4] This program is sold with Microsoft Windows and has just been used to transfer graphics to a format suitable for Nexpert Object.

[5] pLogic Knowledge Systems, 23133 Hawthorne Boulevard, Third Floor, Torrance, CA 90505, Tel: (213) 378-3760.

Software :

. MS-DOS 3.3 or higher operating system with Microsoft Windows version 3.0 or higher,

. the runtime version of Nexpert Object version 2.0 or higher (far less expensive than the development version presently used),

. Microsoft Excel version 3.0, and

. all the remaining software will be compiled code generated in this project.

It is thus expected that most of the software and hardware may already be present in the end-user system, with the exception of the runtime version of Nexpert Object. Indeed Microsoft Windows and Excel are very well known products which can be used for a great variety of applications. This should contribute to lowering the final cost of the expert system.

2.4 Nexpert Object

2.4.1 Introduction

The key choice among the software listed above is that of Nexpert Object[6,7] as the expert system shell. This

6 For a quick review of Nexpert Object features see [Aiken, 1990] or the very good "Introduction Manual" in [Neuron, 1991].

7 The author of this thesis has learned much about Nexpert Object thanks to Neuron Data's Bulletin Board System (BBS), located in California, on which all the Nexpert Object users can exchange tricks and advice.

software has been chosen for its powerful features as described in Dehan's thesis (see [Dehan, 1989]). Thus, its state-of-the-art programming method called OOP (Object Oriented Programming) to assist the developers in economically representing the knowledge and its ability to have access to various databases have made the difference in comparison with lower price commercial shells. In fact, Nexpert Object appeared to be the only PC-based commercial shell including all the features required for this project[8].

We also note that Nexpert Object is rapidly becoming the expert system standard in the AI industry. It already dominates it, comprising 40% of all the expert systems installed. The reasons for this popularity are its equivalent functionality compared to high end, Lisp-based tools at a fraction of the cost[9]. Also, Nexpert Object's visual, menu/mouse driven programming environment simplifies entering, editing and maintaining the knowledge base.

Finally, its open architecture allows the developer to completely embed Nexpert Object into all existing or to be

---

[8] It is interesting to note that although this choice was now made almost 3 years ago, the development of Microsoft Windows and Nexpert Object have only improved this decision. It all works better than originally expected.

[9] Nexpert Object is "C" based, and is cross-compatible on various platforms. It can run on a number of different machines including the IBM AT, PS/2, 286, 386, Macintosh II and SE, Sun and Apollo workstation, DEC Vax with VMS (Nexpert Object can be installed on any VAX station, and support the two windowing systems, UIS and DECwindows) and IBM mainframes with VM. It is compatible with many different operating systems including DOS, OS/2, Mac_OS, Unix, VMS and VM. This interoperability (Nexpert is written in C) ensures that user-developed knowledge bases will run on a variety of products as well as subsequent releases of Nexpert.

developed conventional programs or application using its "AI C library". All the knowledge processing capabilities are accessible through the Nexpert Object Callable Interface[10,11]. They can be embedded in any conventional language : C, FORTRAN, Pascal, C++ ... For example, at runtime, conventional application programs can dynamically create and modify object structures and control the reasoning process. Any existing program written in a conventional language can become a method[12] of an object. By means of attached methods, an object can trigger external functions or processes and interact with the reasoning process. An object in Nexpert Object can represent a database record and its fields. Nexpert Object supports reasoning on records retrieved from Sybase, Oracle, Informix, Ingres, DB2, IMs, Lotus, Rdb, dBase III or IV, and Excel database structures, as well as structures for proprietary databases. Databases and spreadsheets are accessed by Nexpert Object rules and objects through standard SQL queries. In fact, any data structure from existing software can be mapped into Nexpert Object and

10 There are two different ways to use this Callable Interface : calling in and calling out as Neuron Data call them. The first ("calling-in") is the ability for an external program (written in "C-language" for example) to have access to the artificial intelligence kernel of Nexpert Object. The second ("Calling-out") refers to the possibility to call external conventional programming routines while processing in Nexpert Object. This second feature has been used in our prototype.

11 In the latest version of Nexpert Object (version 2.0B) this callable interface is now called the Application Programming Interface. See [Neuron, 1991].

12 A method is an Object Oriented Programming technique which allows the programmer to link a piece of code to a specific object or property. For example, methods have been used in our prototype to link the material properties to the correct fields in our materials databases.

immediately reasoned upon. These features have been used extensively in our prototype to retrieve data from Excel spreadsheets.

2.4.2 Nexpert Object's software description

The package we have purchased includes the following components :

- the development system for use under Microsoft Windows,
- a runtime licence for delivery under DOS (Protected Mode), and
- a runtime licence for delivery under Microsoft Windows.

Our prototype has been developed using the development system which runs in real mode[13] and thus uses the default user interface programmed into this development system. This explains why our prototype should not be seen as representative of the final user interface. This software also includes a bridge to access Excel through the Sylk file format.

---

[13] There are two different modes to program a 386 compatible computer : real or protected mode. The real mode has been designed to assure the compatibility on this new platform with all the existing softwares designed for 8088 or 8086 machines. Thus, real mode keeps the old memory management system and so faces the problem known as the "640 KBytes barrier". The protected mode has been specially designed to take full advantage of the power of the 386 chip and runs using the full 32 bit address space thus giving access to up to 4096 MBytes.

In the next chapter we will see how we have integrated all this different software into our prototype. We will first focus on the flow of the data between the different components of our system, then present our "superset" of data to solve the problem of dealing with different X-ray machines.

3. SYSTEM OVERVIEW

Everybody experiences far more than he understands. Yet it is experience, rather than understanding, that influences behavior.
Marshall McLuhan.

MIT researchers today announced the discovery of a new fundamental particle that is the basis for the widely-known phenomenon of information. Dubbed the "truon", this particle is vital to interactions involving the basic force of truth.
Michael Travers.

We want to fashion puppets that pull their own strings.
Ann Marion.

We have carefully designed the expert system to be independent of any particular hardware implementation of the X-ray residual stress technique. Therefore, our software should be applicable to virtually any system, portable, stationary, manual or completely computer controlled. In this chapter, we outline the basic components of the system and the methods by which these components are linked. The details of the implementation are the subjects of latter chapters.

3.1 Introduction

The realization of our Expert System is embedded in the general procedure of measurement of residual stresses by X-ray diffraction as summarized in Figure [3-1], page 27.

Using this figure we can see nine different steps for such a measurement; four of them already exist, and five more have to be created :

As shown in Figure [3-1], page 27, steps 1 through 4 are the responsibility of the residual stress instrument. They are generally performed on a computer different from the one used for the expert system.

. Step 1: The computer controlled X-ray diffractometer performs the stress measurements on crystalline materials,

. Step 2: The results are sent to the computer[1] which is directly connected to the X-ray diffractometer,

. Step 3: The computer acquires and manages the data and performs data analysis, and

. Step 4: A machine specific "report" is printed.

---

[1] The computer connected to the TEC X-ray Machine in our laboratory is an LSI11/23-S manufactured by Digital Equipment Corporation. It is not compatible with an IBM PC. This divergence of equipment from that proposed in the previous chapter for the implementation of our expert system is typical of the situation regarding most other X-ray stress analysis systems.

These data are also generally logged on a floppy or hard disk, with each stress measurement occupying one filename. Clearly, in a laboratory in which thousands of stress measurements are made annually, file and data management become a major problem. This is especially true in a setting in which many different users are gathering data on the same hardware.

The intermediate steps, the transmission of data from the stress analysis system to a spreadsheet and from the spreadsheet to the expert system (steps 5 and 6) has been designed as the most effective and user friendly method to match our expert system to virtually any X-ray stress analysis system available. In order to move data between the stress analysis instrument and the expert system, any one of several methods may be implemented in steps 5 and 6.

- Step 5: Transmission of the data to an Excel sheet on compatible 386 PC, this can be done by:

  - manually (re)typing the data into a blank sheet[2],

  - scanning a typical X-ray printed output[3],

---

[2] This solution has been used to transfer the first measurements, before we developed an electronic link. We also used it to transfer several "old" X-ray stress measurements for which no source files were available in the PDP 11.

[3] This solution has been tried in our specific case, but since our PDP 11 was connected to a dot matrix printer, an optical character recognition of an output typically produced 10 to 12 mistakes per measurement thus making this solution both tedious and unreliable.

- electronically linking the two computers, or

- transferring an (ASCII) file generated in the X-ray system.

. Step 6: Importation of the Excel sheet into Nexpert Object.

In order to preclude the need for custom programming of our TEC system, as well as the annoyance and inconvenience of retyping data, we have developed an LSI11/23S-to-PC link which captures the printer output of the TEC system and converts it to an appropriate Sylk file for importation to Excel. This system has been working perfectly well during the development of our prototype. But to manage the large amounts of data generated by modern, computer-controlled stress analysis systems we have developed a database management system using dBase III and Clipper 5.0 (see [Courtney, 1991]). We will come back to this DBMS system in Chapter 5.

As illustrated in Figure [3-1], page 27 it is the function of the expert system to access X-ray residual stress data and materials properties both of which reside in separate databases on the host ES computer, to analyze these data based on rules and other knowledge code in the expert system, and to print a report. These are performed in steps 7 through 9:

. Step 7: Validation and interpretation of the data by the expert system,

. Step 8: Access the materials databases, (also developed as Excel sheets), and

. Step 9: Production of a final report (including all the expertise) following the expert system analysis.

For the initial tests of our expert system, we have chosen to use Excel as a user friendly medium by which data may easily be loaded into the ES. This decision was made for several reasons. First, the link to Nexpert Object was a proven link, and second, the spreadsheet provides a user-friendly medium in which data may be graphed, analyzed and manipulated by users who have no programming experience. Microsoft Excel also supports Dynamic Data Exchange (DDE), which allows two applications running under Windows to exchange data without the need to create temporary files. We note that Microsoft itself has used just such an interface for a major ES application of its own[4] . However, we must also note that this can only be a temporary solution because it does not solve the problem of managing thousands of stress measurement results. To resolve this problem, we have developed a Database Management System (DBMS) based on dBase III DBMS compiled with Clipper 5.0. This system retains the virtues of a powerful, user-friendly DBMS (dBase III) and direct access to the data by Nexpert Object through the use of C-programs to extract data from the DBMS, convert it to the Sylk file format used in Excel and readable by Nexpert Object[5].

---

[4] See [PC Week, 1990].

[5] See [Courtney, 1991] and [Potet, 1991] for more details.

3.2 Discussion and focus of the present work

During the development of the prototype system we have considered step 5 only in the case of the TEC instrument[6] (see the TEC Model 1600 pictured in Figure [3-2], page 28). Indeed, this step is obviously unique to every X-ray machine and must be implemented specifically in every different case. During the testing and debugging of the prototype, data have been transferred from the PDP11 of the TEC machine to the spreadsheet by hand using a blank Excel sheet and more recently with our electronic link. We call this blank sheet a "superset", (see Figure [3-3], page 29) and which will be discussed in more detail in section 3.3.1, page 22. In the production system, this form of interface will prove to have the greatest flexibility to accommodate the widest range of stress analysis equipment. In those cases where the equipment is not computer automated, the spreadsheet provides an easy, user-friendly method by which data may be entered into the system manually. In the case of computer automated system, it will be a simple problem to write a short program, in virtually any high-level language, which outputs the data in a standard ASCII comma delimited file[7] which may be imported directly into the spreadsheet. If certain data are missing from a given machine, the cells are left blank and appropriate action is taken by the expert system.

---

[6] It has to noted that we are in touch with other manufacturers to develop equivalent importation programs for their own specific X-ray analysis system.

[7] As implemented in our system but any other industry standard format can be used.

We have decided that the development of a user-friendly interface to Nexpert Object, as required for step 9, should be the last stage in the development of our software. Indeed, it is obviously dependent on all the results obtained during the expert system process. Thus, we have left this final development for future work and the remainder of this report focuses on steps 6, 7 and 8. However it appears that with powerful features[8] of the current version of Excel we will be able to do this entirely within the spreadsheet.

Step 7 is the heart of the expert system. It is the rule and object-based inference system which performs the data analysis. Step 6 is the method of linking the data to the data input spreadsheet into the system, and step 8 is the development and interfacing of the materials library database required for the expert system. Step 6 has been designed using static importation of data from Excel spreadsheet to Nexpert Object[9], whereas step 8 requires dynamic retrieval.

3.3 Description

In this section we describe how we have solved the problem of importing the data into our expert system from any existing X-ray stress analyzer.

---

[8] And especially the possibility to customize the pull-down menus.

[9] As previously noted this approach has been working very well in the development of our prototype but we are also implementing a more powerful database management system as described in [Courtney, 1991].

3.3.1 Data Superset

In order for our expert system to accept data from any diffractometer available on the market it is essential that all data be transformed into a common format. This has been accomplish by developing a "superset" of data, which can accept any kind of input.

Thus we have written to several experts worldwide to ask them to provide us with the data they think are necessary in the interpretation and validation of measurement of residual stresses by X-Ray diffraction. The experts who have assisted us are listed in the following table:

Mr. M. Brauss	Proto Mfg, Oldcastle, Ontario
Dr. J. Cohen	Northwestern University, Illinois
Dr. V. Hauk	Technishe Hochschul, Aachen, Germany
Dr. M. James	Rockwell International Science Center, Thousand Oaks, California
Mrs. B. Pardue	Technology for Energy Corporation, Knoxville, Tennessee
Dr. P. Predecki	University of Denver, Colorado
Dr. P. Prevey	Lambda Research, Cincinnati, Ohio
Dr. B. Scholtes	Technishe Hochschul, Karlsruhe, Germany

The result of the input of these different experts or manufacturers has given us our current version of the superset which is shown in Figure [3-3], page 29. In this figure all the grey cells can be filled with input coming from the X-ray stress analyzer or from the expertise of Nexpert Object. This format will be of great value in the Expert System report. Indeed we will only need to append our analysis to this standard report completed with a d-vs-$\sin^2$(Psi) graphic. This will constitute the typical output of our expert system.

3.3.2 Development of rule-based knowledge

Data which illustrate several typical problems in X-ray residual stress determination[10] have been submitted to us by several laboratories or obtained from colleagues in the VPI&SU Residual Stress Laboratory. We have started to analyze these data to determine how they may be classified from the point of view of an expert system. The results of these analyses as well as the statistical analysis presented in the Chapter 5 "Statistical Analysis", page 48, form the basis for the rule-based aspects of the proposed system.

From the different data received from various laboratories and experts (see previous section), several rules have been implemented with Nexpert Object. The focus of these rules has been the validation of data (to discriminate between bad and good data) rather than the interpretation, which should be designed in phase III of the work. As was pointed out by Dehan in his Masters thesis[11], the problem of residual stress analysis can be separated into two major components: the validation and verification of the residual stresses as determined by a single X-ray stress measurement, and the interpretation of these stresses in the context of

[10] Such as grain size, preferred orientation, or multiple peak problems.

[11] See [Dehan, 1989], Chapter 2, "Impact Study".

measurements on a wide range of similar or dissimilar components. The prototype system focuses on the former. The latter cannot be addressed until one has a sufficient database of measurements for comparison.

To validate the data, a wide range of expertise and knowledge is required. Much of this information can be written in the form of "if ... then ..." statements. For examples, the following rules have been developed:

- verification of the d spacing value for all the different Psi angles measured,

- verification of the value of the residual stress (in comparison with the materials database)[12],

- identification and classification of the d-vs-$\sin^2$(Psi) curve. This specific problem has required a lot of energy[13] to discriminate among several possible curves (see Figure [5-2] page 67).

In all, several complex rules have been developed which are aimed at insuring the valid interpretation of the data. Again we have to say that these rules do not pretend to solve all the existing problems of validation and

---

[12] As pointed by Dehan, there are three problems in the analysis of a measurement of residual stress by X-ray diffraction: (i) are the data valid ?, (ii) what can we say about d-vs-$\sin^2$Psi for one run ?, and (iii) can we compare a run with lots of other data ? In our work we have focused on (i) and (ii) but have not implemented (iii).

[13] We will come back to this specific problem in the Chapter 5 "Statistical Analysis", page 48.

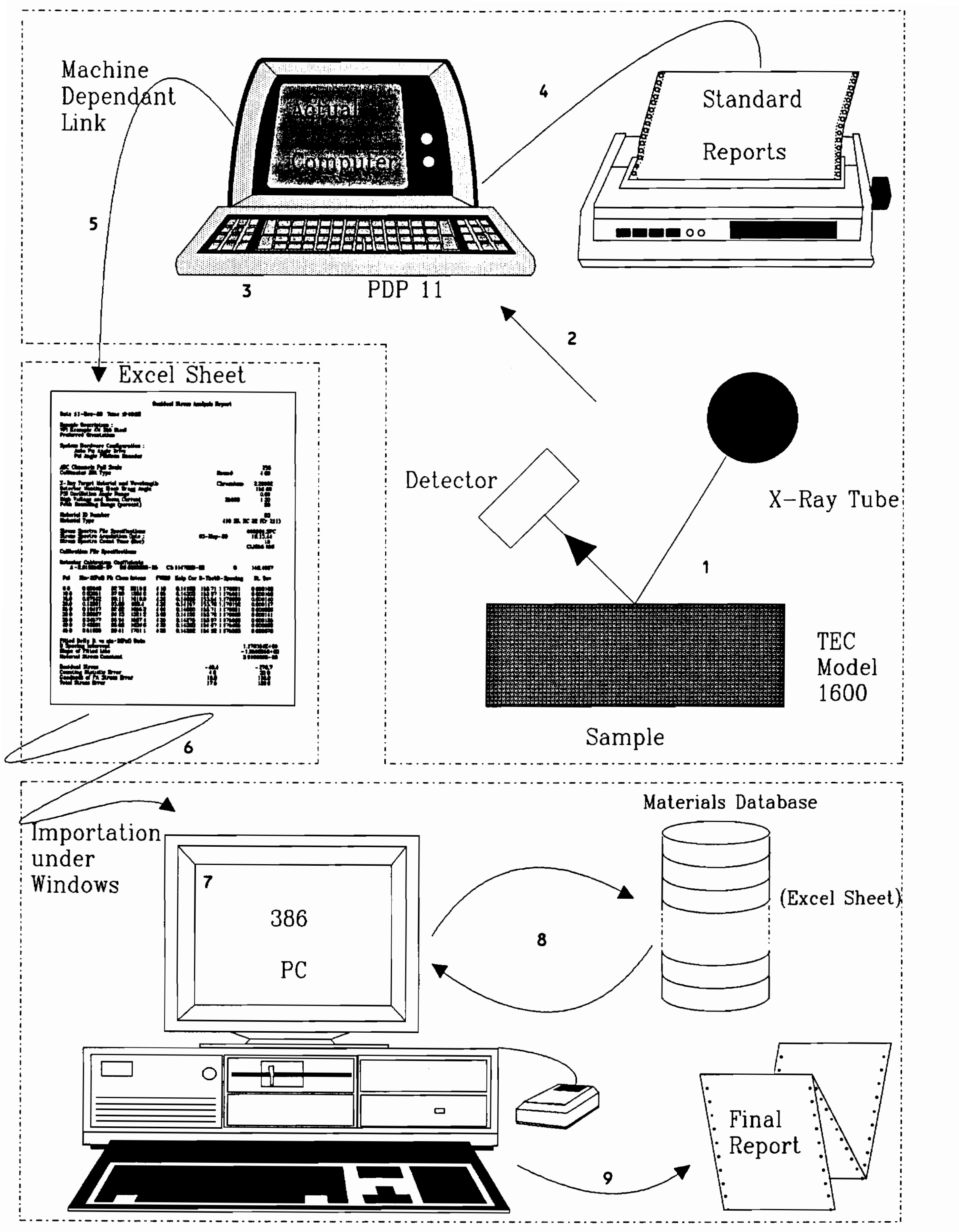
interpretation of residual stress data, but have been chosen as representative of the complexity of the problem. A wide range of rules, such as those concerning grain size problem, tri-axial stresses, or preferred orientation still have our attention and will be developed in the next phase of this project.

3.4 Maintenance

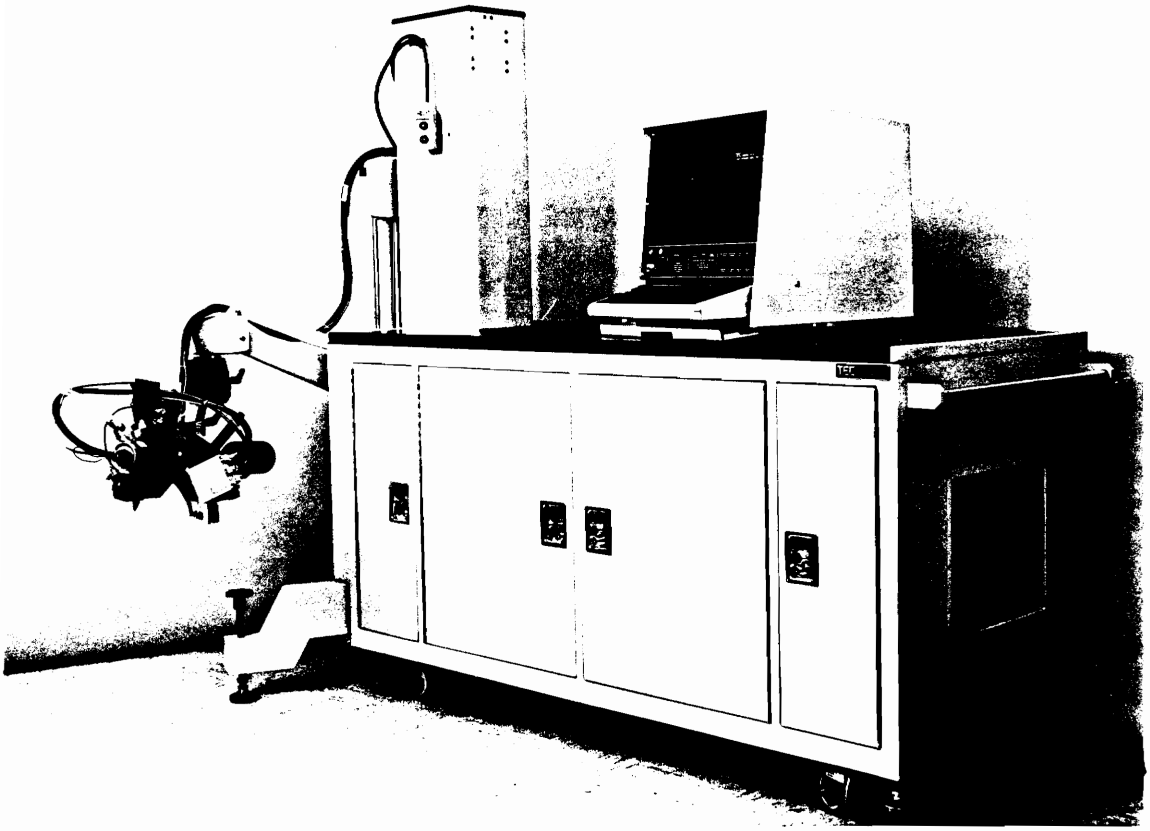
Among the objectives which dictated the choice of the expert system shell is the requirement that the end user may easily maintain and update the system for a maximum variety of materials. Since we cannot expect him (or her) to spend several months learning how to program Nexpert Object, we have designed our materials database as several simple Excel files. Therefore, if the end user must work with a material which is not included in the present databases, all he will have to know is how to use the Excel spreadsheet, and thus will be able to update the databases adding his new material characteristics in just a few minutes following the examples provided.

As described above, the same concept is true for importing data from the X-ray machine. Therefore, if a user has to interpret the data from a machine for which no specific driver has been written yet, he can just modify data manually in an existing sheet or fill a new one using the blank sheet provided.

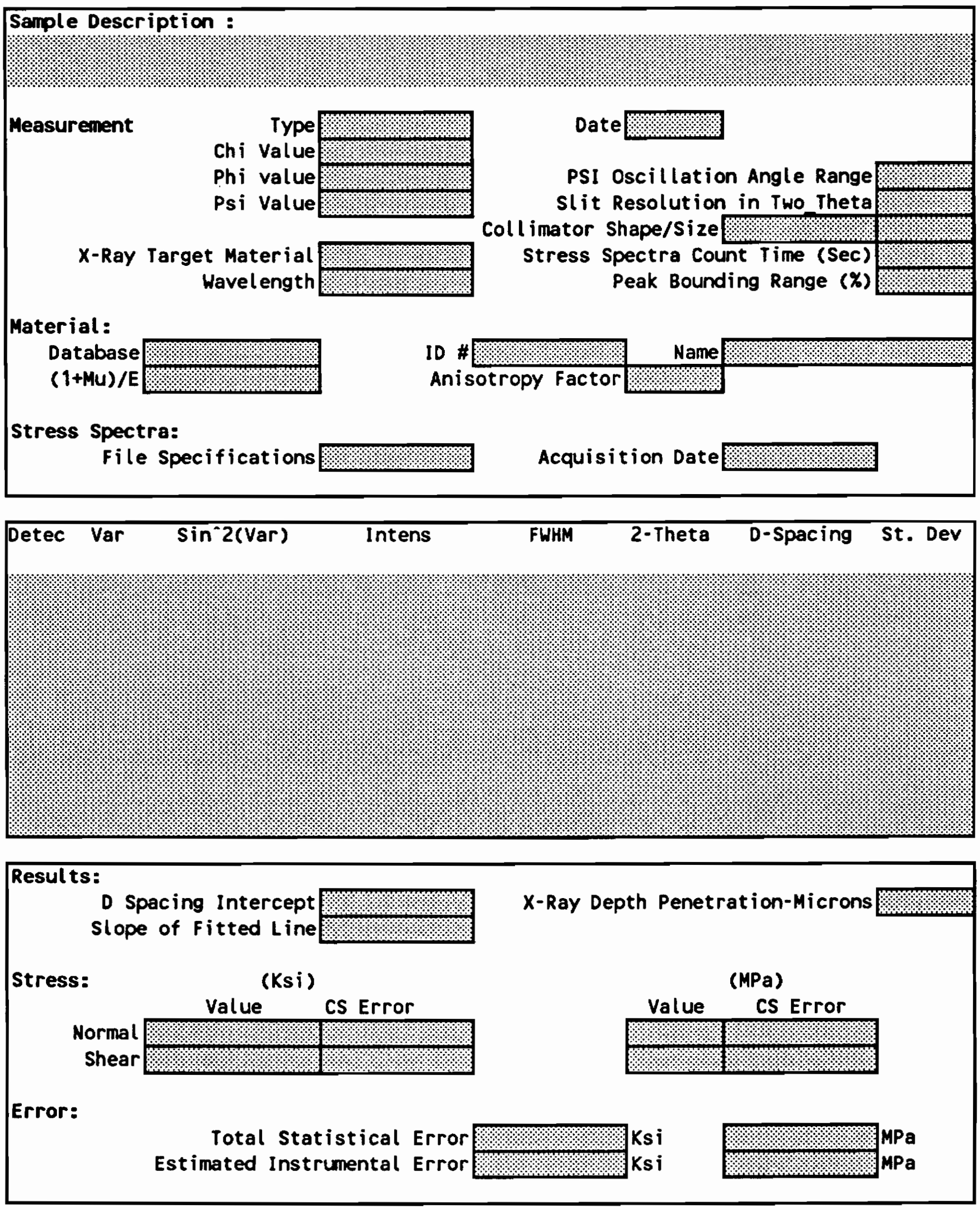
In the next two chapters we describe how we manipulate the data in our expert system. Chapter 4 will first present the utilization of the power of Nexpert Object for the manipulation of qualitative reasoning during the validation part of our expertise. Then Chapter 5 will describe the statistical and pattern recognition techniques used in our attempt to discriminate and sort our data.



SYSTEM OVERVIEW - Figure [3-1] General overview



SYSTEM OVERVIEW - Figure [3-2] TEC Model 1600



SYSTEM OVERVIEW - Figure [3-3] Superset of data

4. KNOWLEDGE ACQUISITION AND REPRESENTATION

All virtue is one thing, knowledge.
Plato

Knowledge is not the same as information. Knowledge is information that has been spared, shaped, interpreted, selected, and transformed; the artist in each of us daily picks up the raw material and makes of it a small artifact - and at the same time, a small human glory. Now we have invented machines to do that, just as we invented machines to extend our muscles and our other organs. In typical human fashion, we intend our new machines for all the usual purposes, from enhancing our lives to filling our purses.
Edward A. Feigenbaum and Pamela McCorduck, *The fifth generation*.

Birds can fly, unless they are penguins and ostriches, or if they happen to be dead, or have broken wings, or are confined to cages, or have their feet stuck in cement, or have undergone experiences so dreadful as to render them psychologically incapable of flight.
Marvin Minsky, The Society of Mind (illustrating the difficulty of accurately expressing knowledge).

Knowledge is power and permits the wise to conquer without bloodshed and to accomplish deeds surpassing all others.
Sun Tzu, *The Art of War*.

In this chapter we focus on the representation of the knowledge into our expert system. We have carefully designed our expert system to take advantage of the full power of Nexpert Object, which allows the use of "if ... then ..." rules as well as object-oriented techniques.

4.1 Knowledge acquisition

4.1.1 Introduction

According to various knowledge engineers, elicitation of expert knowledge can be done using three different methods[1]:

. by analyses of the tasks that experts perform

The tasks usually performed by experts are charted, step by step, and are analyzed at whatever level of detail is sufficient for the purpose of the analysis. Such a task analysis can involve direct observation or can come from documentation such as job descriptions, textbooks, open literature and articles, etc...

. by interview techniques

This method, which has been chosen by many, if not most, system developers, consists of obtaining the expert knowledge by conducting a series of detailed question and answer interviews. These interviews can be unstructured (in the form

[1] See [Hart, 1985], [Hoffman, 1987], [Hoffman, 1989], [Mittal, 1985], and [Prerau, 1987].

of a free dialog in which open-ended questions are asked about the expert knowledge) or structured (in the form of detailed questions prepared in advance).

. by special tasks

This method relies on techniques that have been used in experimental psychology, such as the protocol analysis method, which originated in the study of problem solving.

4.1.2 Chosen method

We have used the two first methods. The task analysis was done using various textbooks[2,3,4] and the TEC operating manuals[5]. The interviews (of Dr. Hendricks) were done in a printed question-answer form.

---

2 [Noyan, 1987].

3 Advances in X-Ray Analysis, Plenum Press.

4 [Castex, 1981].

5 [TEC, 1980].

4.2 Knowledge representation

4.2.1 Introduction

Contrary to most of the low level expert system shells available today[6,7,8], which are only able to generate rules, Nexpert Object[9] is an hybrid system which allows a knowledge representation using both rules and objects and thus has, in addition, full access to the power of the object oriented programming (OOPs) concepts. For a very good introduction of these different concepts see the "Introduction Manual" in [Neuron, 1991].

4.2.2 Rules

4.2.2.1 Specific design

Nexpert Object uses a specific syntax for its rules called the "augmented rule format". In this syntax the programmer does not need to define a rule as "forward" or "backward"[10]; indeed the inference engine will process the rule in either direction as needed.

---

6 For a quick description of several expert systems shells see [Myers, 1986].

7 See [Gilmore, 1985].

8 See [Gilmore, 1986].

9 It is interesting to note that the original version of Nexpert delivered in 1986 was also only a rule-based system and it was not until the delivery of Nexpert Object in 1988 that the OOP technique was added.

10 For an brief explanation of these AI techniques see the Paragraph 4.2.4, page 40.

Typically a Nexpert Object rule has the following format:

IF ...

Condition 1	**THEN**	Hypothesis
Condition 2		
...	**AND DO**	Action 1
...		Action 2
Condition N-1		...
Condition N		Action M

For example the rule to verify the value of the residual stress obtained by comparison with the value included in the Materials database would be written:

- In pseudo code:

  If "all the stuff having to be done is finished"

  And "the computed stress value is greater than the acceptable value given in the materials database"

  Then "the stress value is too big"

. In Nexpert Object high level code[11]:

```
RULE :  Rule
Stress_Value_Verification_Rule (#9)

If        there is evidence of
Statistical_Evaluation _Ready

And       'Material_'\Material_Id_Number
\.Max_R_S*'Material_'\Material_Id_Number
\.Yield_St/100 is assigned to
'Material_'\Material_Id_Number\.Max_R_S

And       ABS(Residual_Stress)-
'Material_'\Material_Id_ Number\.Max_R_S
is greater than 0

Then      Stress_Value_Too_Big is
confirmed.
```

The comprehensive list of the rules which have developed in our expert system can be obtained from Dr. Robert W. Hendricks in the Department of Materials Engineering at VPI&SU.

4.2.2.2 Rule network

Nexpert Object links rules (and objects, as will be discussed in the next section) in a network which allows the deduction of information. A simple example is given in

---

[11] Nexpert Object has two levels of representation for its knowledge base, a low level used by the developer during the coding phase and a high level which is more readable.

Figure [4-1], page 44.

In this example we see that there are three separate rules:

RULE : Rule 1
If there is evidence of A
And there is evidence of B
Then C is confirmed.

RULE : Rule 2
If there is evidence of D
And there is no evidence of E
Then F is confirmed.

RULE : Rule 3
If there is evidence of C
And there is evidence of F
Then G is confirmed.

These three rules are linked[12] by the inclusion of the results of the first two in the third. In this example in order for rule #3 to be fired, the rules #1 and #2 have to be examined first.

With Nexpert Object the rules can have four different statuses during the knowledge processing, these four statuses are:

- Unknown: the rule has not been evaluated yet,

---

[12] In this case Neuron data talks about a "strong link" between the rules 1 and 3 for example since the hypothesis of the rule 1 ("C") is also used in rule 3. The other example of a link (a "weak link") will be seen further.

- False: the rule has been evaluated and found to be false (i.e at least one of the elements of the "If part"[13] was false),

- True: all the components of the "If part" were found to be true,

- NotKnown: the rule has been evaluated but there were not enough data to draw a conclusion.

4.2.3 Objects

4.2.3.1 General design

As its name implies, Nexpert Object is object-oriented in nature and thus allows a representation of data using classes, objects and properties[14]. Furthermore, inheritance can be used to access information.

An object is an elementary unit of description, a property is one of its characteristics and a class is a collection of objects which share properties. A class of objects has a variety of properties. Inheritance means that if an object belongs to a class some or all of the properties of the class are automatically assigned to the object. Specific values of a property can be assigned either for all objects in the class or independently and uniquely for each object.

These concepts are illustrated in Figure [4-2], page 45. In this figure a class ("Class") has been defined with three

[13] Neuron Data refers to this as the Left Hand Side (LHS).

[14] For an introduction to object-oriented techniques see [Fikes, 1985] or [Stefik, 1986].

properties. This class also has two different objects (Object_1 and Object_2) which inherit the three properties previously defined. It should be noticed that Object_2 also has a property ("Specific_Property") of its own.

Using these concepts, one can write "generic rules" for a class which can be executed for all of its objects. This constitutes one of the most powerful features of Nexpert Object.

4.2.3.2 Object network

An example of graphical representation of the classes, objects, properties and MetaSlots (to be discussed in the next section) created for our prototype can be found in Figure [4-3], page 46. As shown in this figure the class "Materials" does not contain any objects initially. It is composed of the fifteen metallurgical and mechanical properties needed in our analysis. Since we obviously do not know in advance the material that is going to be analyzed, the object corresponding to this material will be created dynamically during the session. This object then will be linked to the class "Materials" and therefore inherit all its properties, as well as the methods describing how to get access to the specific database to find the desired value. The resulting object structure is shown in Figure [4-4], page 47.

This object network is very useful for displaying a comprehensive list of objects and/or properties and/or classes with or without their MetaSlots. It has to be noticed that this object structure matches perfectly the design of our "superset of data" presented in section 3.3.1, page 22.

This entire process is described in more detail in paragraph 6.3.3, page 94.

4.2.3.3 Inheritance

In Nexpert Object (as in most of the Object-oriented languages) objects may inherit properties from other objects or classes. This inheritance can be controlled by Nexpert Object and can even be changed during a process at any time. This feature has been used for example during the dynamic information retrieval from Excel: a class ("Materials") has been designed with all the characteristics (both metallurgical and mechanical) which are required to describe the material from the X-ray residual stress point of view. A MetaSlot[15] has been designed which accesses the necessary materials properties from several Excel spreadsheet databases as they are needed during the analysis. The dynamic creation of an object (Material_83, for example) linked to this class "Materials" triggers the inheritance of this routine.

The concepts of MetaSlot, as implemented by Neuron Data in Nexpert Object, is perhaps one of the most powerful

[15] A MetaSlot in Nexpert Object is a method linked to a property and which contains two kinds of information: "Order of Source" and/or "If Change". The Order Of Source (used in our particular example) links a routine to the property which is to be run during the first utilization of this property. The If Change MetaSlot is used to trigger a subroutine whenever the value of the property under consideration changes. This is particularly useful for real-time applications. For example, in real-time process control an If Change MetaSlot may be linked to the property Boiler.Pressure and trigger an alarm if the pressure is more than 2000 Psi. See section 6.3.3, page 94 for an example of MetaSlot used in our system.

techniques available in expert system technology for dynamically loading, changing, or checking data during the processing of the rules in the knowledge base. We have used this technique extensively in developing the prototype system described here[16].

4.2.4 Reasoning

Although most expert system shells provide only two methods of inference (i.e process) on rules (forward and backward chaining)[17], Nexpert offers a total of four :

. forward chaining

This is a data driven method; the process starts with data available and determines related conclusions. These conclusions are then added to the data, and the process starts again.

. backward chaining

This is a goal driven method in which the process starts with an hypothesis and determines the data necessary to conclude the value of the hypothesis (as TRUE, FALSE or NOTKNOWN)

---

16 In fact, the author had to rewrite the entire knowledge base to take advantage of this powerful technique after participating in a one week training program at the Bechtel AI Institute.

17 For example the author has programmed with ART-IM, developed by Inference Corporation, another powerful shell available on the market. ART-IM is essentially Data-Driven (i.e use forward chaining) and backward-chaining can only be implemented with tricky procedures.

. forward chaining thru gates

This method is applied between rules which share at least one datum in their conditions. It means that evaluating a condition in one rule can fetch data which causes forward chaining to other rules.

. context links

Special links (called "weak links" in Nexpert Object as opposed to "strong links" between rules which share at least one datum) can be programmed by the developer. This concept of weak links can be used for example in what is called a "Knowledge Island" by Neuron Data[18].

It should be noted that all these inference modes can be mixed (and in fact that is the default strategy in Nexpert Object) during a session.

4.2.5 Example: Representation of our superset

In this section we present the use of object oriented programming techniques to represent our superset of data as defined in the previous chapter. From Figure [3-3], page 29, we can see that our superset has been divided into three different parts:

---

18 A knowledge Island in Nexpert Object consists of several rules related by strong links only i.e. sharing the same hypothesis for example. These knowledge islands can also be loaded separately in Nexpert Object and so can be seen as overlays in conventional programming. The term "island" comes from the graphical representation of such rules, in which they appear to be isolated, from the rest of the knowledge base, in the rule network.

- the upper part describes the experiment,

- the middle part is the d-vs-$\sin^2$(Psi) itself with all the associated statistics, and

- the lower part which represents the results of the experiment.

In the definition of our object hierarchy we have represented the same structure. Indeed our superset is represented with three classes called "descriptions", "angles" and "results"[19]. These classes contain objects and properties which perfectly match the fields of our superset. For example the class "descriptions" contains four different objects called:

- "Material" with 5 properties,

- "Measurement" with 14 properties,

- "Sample_Description" with 2 properties, and

- "Stress_Spectra" with 3 properties.

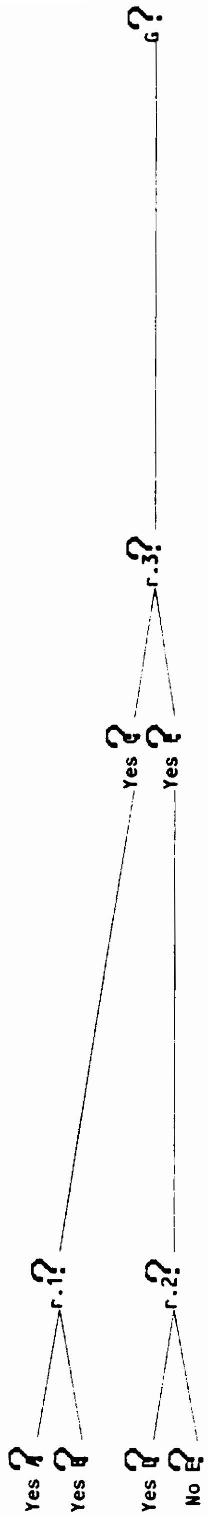
As it can be seen from Figure [3-3], page 29, the number of properties linked to any of these objects correspond to the number of grey cells in our superset, i.e. the number of data fields.

---

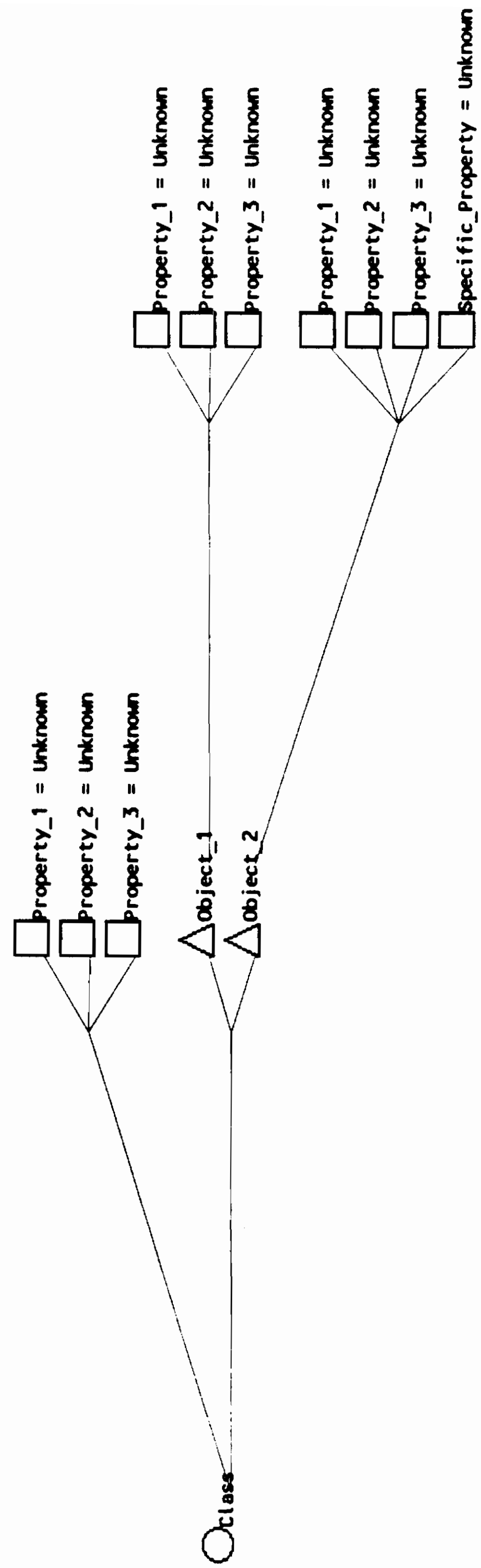
[19] We have chosen to represent all the classes with a name ending with an "s" as opposed to the objects or properties whose names end without it. For example we have the class "Materials" with an "s" and the object "collimator_Shape" or "Psi_Value" without "s". This is a very common practise in object-oriented programming.

4.3 Conclusions

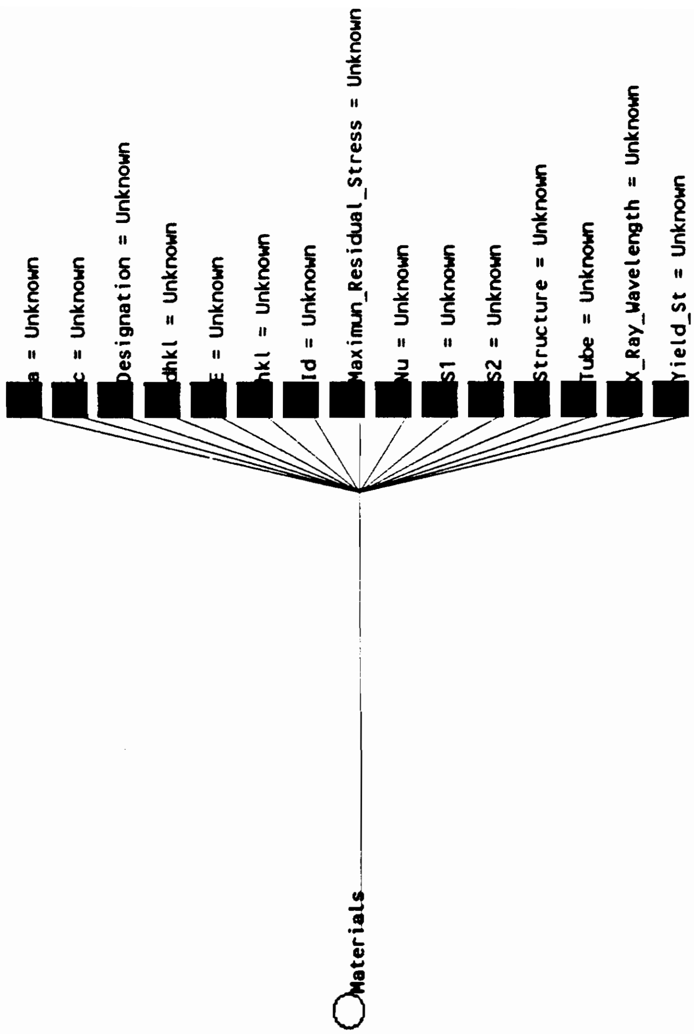
As the rule-based expert system was being
developed from the expert knowledge, it became clear that
such a … e to describe the
expert's … in analyzing and
interpre… was found that there
were maj… ognition process. Much
of the … e shape of the d-vs-
$\sin^2$(Psi … to think in terms of
develop… extended statistical
analysi… eeded to be performed
prior t… sis in Nexpert Object.
The g… oject provided such
capabil… the next chapter.



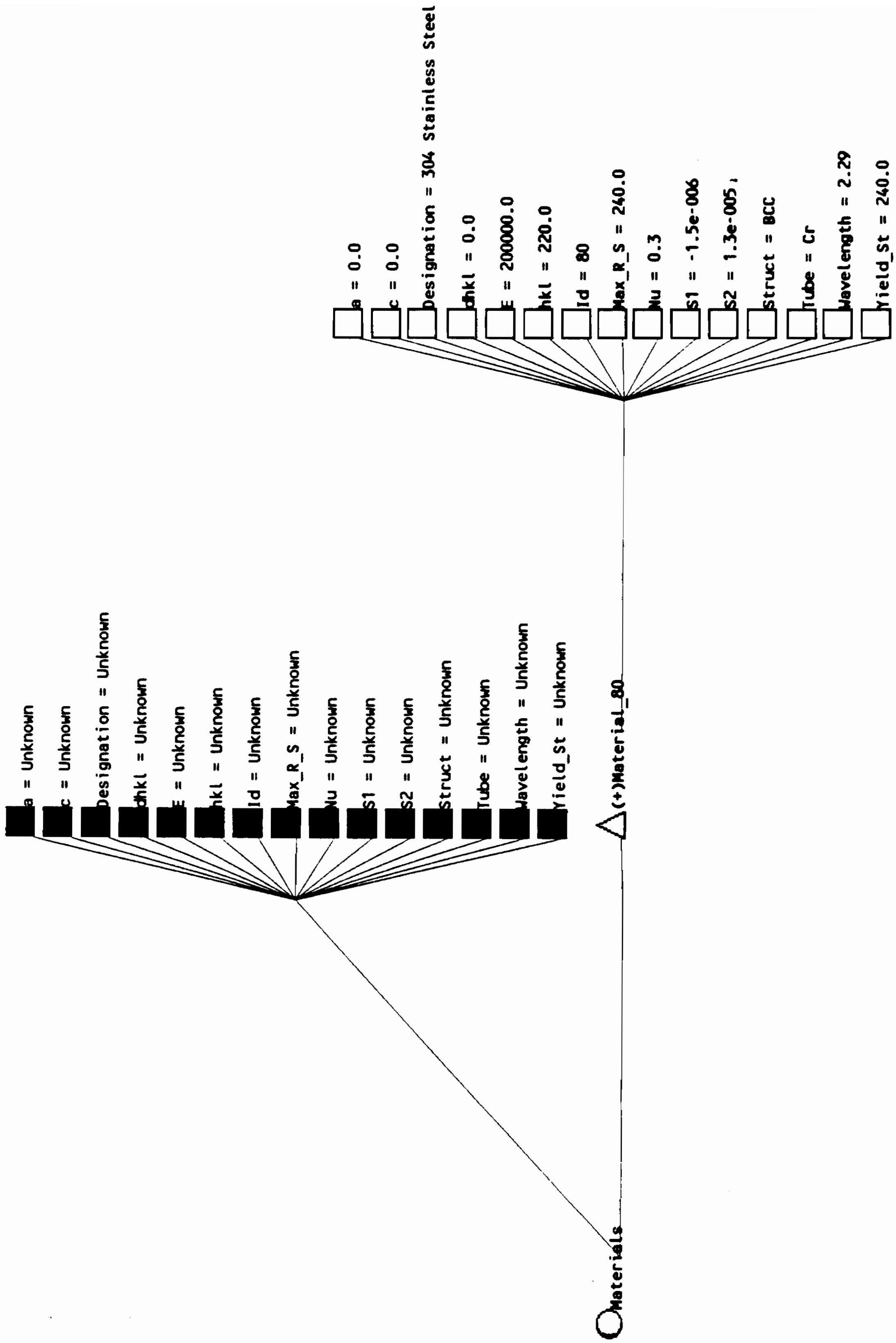
KNOWLEDGE ACQUISITION - Figure [4-1] Rules example



KNOWLEDGE ACQUISITION - Figure [4-2] Objects example



KNOWLEDGE ACQUISITION - Figure [4-3] Class "Materials" (before importation)



KNOWLEDGE ACQUISITION - Figure [4-4] Class "Materials" (after importation)

5. STATISTICAL ANALYSIS

A Mathematician is a machine for turning coffee into theorems.
Paul Erdos.

Politics are for the moment. An equation is for eternity.
Albert Einstein.

The machines will get good enough at dealing with complexity that they can start dealing with their own complexity, and you will get systems that evolve.
Daniel Hillis.

One of the most complex problems facing us in the development of our expert system is the classification of the data in order to apply specific treatments according to the expert's understanding of the results.

5.1 Introduction

We have started a statistical analysis[1] to classify the data obtained during a measurement. Indeed as it appears in Figure [5-1], page 66, the analysis of measurement is composed of four different stages:

- First, the data are imported from an X-ray stress analyzer or from an X-ray database. This problem is purely a "computer science" problem and has been solved as shown in the previous chapters.

- Then, the data must be validated. That is, we must be sure that our measurements have been performed correctly before going any further in the analysis. This problem has been solved using Nexpert Object and applying several significant rules to verify point by point every detail of the measurement.

- Third, the validated data must be classified. This is the goal of the statistical procedures described in this chapter.

- Eventually, according to the various categories of d-vs-$\sin^2$(Psi) curves (bi-axial stress, tri-axial stress, preferred orientation, grain size effects and/or stress gradients ...) as previously determined, a specific statistical procedure must be applied. This analysis has not yet been incorporated into our system, but appears to be just a question of code writing.

---

1 See [Ott, 1988], [Schulman, 1990] and [Schulman, 1991].

It should be noticed that this four-step approach corresponds to a human analysis of data. Indeed, as the author has been able to determine through his interviews with Dr. Hendricks, an expert:

- Takes the output from an X-ray stress analyzer (Step 1),

- checks for any data inconsistencies, scanning for some typical mistakes , such as abnormal intensities, counting statistics errors, and d-spacings (step 2),

- examines the d-vs-$\sin^2$(Psi) curve to determine if the bi-axial assumption usually used in the stress analyzer is valid or not, and

- if not, attempts to classify the curve into one of the other groups identified in Figure [5-2], page 67. Based on this classification, a recommendation is made regarding how one should handle and/or interpret the data.

But, as we will see in this chapter, the third step (recognition of the type of curve) is not a trivial problem.

5.2 Compilation of data

In order to proceed with the statistical analysis we have built a database of several hundred real measurements on various materials (stainless steel, aluminum, Incoloy, nickel...) made in several residual stress laboratories during the past few years[2]. These data include examples of almost every typical problem encountered in X-ray analysis.

5.3 Classification of data

Based on the experience of our human expert we have recorded that most of the data obtained from the precedent compilation can be classified into five different categories. This analysis is based on a comparison of the data obtained with well known models representative of various typical problems (see Figure [5-2], page 67).

5.3.1 Case I

This first case is the easiest one. It consists of a linear d-vs-$\sin^2$(Psi) plot. This normally corresponds to a biaxial state of stress in the surface of the sample and is used as the default assumption in most commercial stress analyzer instruments. If the expert recognizes this specific shape of the curve, he knows that the stress value estimated by the X-ray stress analyzer is fairly accurate.

The equation of this curve can be written:

$$d = A + B \sin^2(Psi)$$

where:

. d = d-spacing of the material, and

---

[2] We are indebted to Dr. J. Jo, V. Iyer, N. Schulz and E. B. Pardue for providing examples of "difficult" data for this analysis.

. A and B are constant terms which include the X-ray elastic constants of the material and the normal stress components of the stress tensor.

5.3.2 Case II

The second shape consists of a portion of an ellipse. It corresponds to a tri-axial state of stress in the surface layer of the material. In addition to the normal stress components of the previous case we also have a shear stress component.

The equation of this curve is:

$$d = A + B \sin^2(Psi) + C \sin(2Psi)$$

where:

. d = d-spacing of the material, and

. A, B and C are constant terms.

It should be noticed that in this case we can distinguish between the positive Psi values (represented with squares in Figure [5-2], page 67) and negative Psi values (represented with circles in Figure [5-2]) because Sin(2Psi) is an odd function in Psi while Sin^2(Psi) is an even function.

5.3.3 Case III

The third kind of shape of d-vs-$\sin^2$(Psi) appears in the case of strong preferred-orientation and/or grain size in the material. Although progress is being made in this direction (see [Castex, 1981]) there is so far no reasonable analytical expression for this kind of curve, thus making its analysis beyond the range of our expert system. However, it is still essential that the system be capable of recognizing such patterns. As will be seen, we are making progress in handling such data through a process of elimination.

5.3.4 Case IV

This shape curve (referred in our laboratory as a "fish curve") randomly appears in some measurements we have performed in our laboratory. Even if no theoretical model has been developed so far for this specific kind of curves, it has been found that it may fit an equation of the form:

$$d = A + B \sin^2(Psi) + C \sin(2Psi) + D \tan(Psi)$$

where:

- d = d-spacing of the material, and
- A, B, C and D constant terms.

It is believed that such curves may be a result of X-ray optics errors and/or sample misalignment.

5.3.5 Case V

On occasion one observes curved d-vs-$\sin^2$Psi plots. This curvature is attributed to variations in the stress as a function of depth below the surface, and has been modeled mathematically (see [Noyan, 1987]).

All of the data we have collected have been classified according to these five categories. A summary of this classification is give in Figure [5-2], page 67.

5.4 Data Analysis

Following classification of the data by human expertise, two different approaches were taken to determine if such classifications could be performed by computer analysis. First, it was recognized that the human data classifications were actually an unconscious form of qualitative pattern recognition. Did the data fit one of the analytical models or not? Therefore, it seemed reasonable to believe that such classification might be performed by a regression analysis of the analytical expressions and an examination of standard F-tests and t-tests for significance of fit. Second, it became apparent that the problem might be sufficiently complex that the techniques of statistical pattern recognition might be more appropriate. Each of these techniques is described in more detail in the following sections.

5.4.1 Regression Analysis

5.4.1.1 Introduction

The first approach used to discriminate data has been a classical multiple linear regression. For each of the previous four cases we have taken a few significant examples of real data and fed them into a statistics package[3] to perform multiple linear regression. We have also computed statistical F-tests to determine if a measurement is best compared to a specific model, as well as t-tests to determine the significance of the regression coefficients. Thus, all the classic d-vs-$\sin^2$(Psi) curves have been

[3] We have used CoStat, sold by Cohort Software, P.O. Box 1149, Berkeley, CA 94701.

related to a straight line, an open curve, a fish curve and a non-linear curve.

Figure [5-5] and following starting on page 70 show several examples of such analyses. In all of these figures the following procedure has been applied:

- First, we start with an output obtained from a real measurement with our X-ray diffractometer. Figure [5-3], page 68 shows this output for the example of the experiment "604".

- From this output we extract the columns:
  . Psi,
  . $Sin^2(Psi)$, and
  . d_spacing.

- These columns are then used to compute the following additional columns:
  . Psi_Real[4],
  . Sin(2*Psi), and
  . Tan(Psi).

At this point the data obtained from measurement "604" can be seen in Figure [5-4], page 69.

---

[4] Indeed the Psi angle value used in the TEC diffractometer we used is specified by the user before the real experiment. The Psi angle value which will really be used during the experiment ("Psi_Real") will be slightly different. The Psi-Input (representing the calculated Bragg-angle) is located at the center of the detector. In reality the peak position is not necessarily at this position. Thus the "Psi_Real" has been calculated from the $Sin^2(Psi)$ column with the formula:

$$Psi\_Real = ASIN(SQRT((Sin^2(Psi))) * SIGN(Psi)$$

- A "backward multiple regression" can then be computed. Such a multiple regression consists of the following phases:

  . First, we try to fit the full model into our data, that is the equation with the $Sin^2(Psi)$, Sin(2*Psi) and Tan(Psi) terms as it appears in Section 5.3.4, page 53.

  . In Figure [5-5], page 70, and Figure [5-6], page 71, we can see this first regression. It should be noted that our software provides the R^2 value for the fit, as well as the F-Test and p-value[5].

  . The least significant variable[6] is dropped if it is determined to be not significant, and is started the regression again with the remaining two parameters. This process is iterated until all coefficients are significant or until the system fails to find any fit at all.

This procedure has been used for several dozens of measurements, which were then compared with a "manual classification" performed by Dr. Hendricks. The next sections present four different examples of such analysis[7].

---

5 See [Schulman, 1991], or [Ott, 1988] for an explanation and a discussion of the significance of these different statistic coefficients.

6 Shown with a "ns" in the rightmost column in Figure [5-5], page 70.

7 We have not considered in the following examples the Case V described previously. Indeed, at the time of this analysis we did not have enough data which fell in this specific category.

5.4.1.2 Example of Case I

This first example[8] (shown in Figure [5-5], page 70) presents an example of measurement which has been interpreted as Case I by Dr. Hendricks. As explained previously our statistical software first ran the full model (with the $Sin^2(Psi)$, Sin(2*Psi) and Tan(Psi) terms), and obtained:

. $R^2 = 0.9828$,

. F = 57, and

. P = 0.0038 (that is 0.38 %).

Thus, this first regression is seen to be globally significant, but the software suggests that two parameters (the Sin(2*Psi) and Tan(Psi) terms) are not significant. Thus a new regression is repeated with only the $Sin^2(Psi)$ and Sin(2*Psi) terms. That is we have dropped the Tan(Psi) term, which the least significant one. The result is:

. $R^2 = 0.9819$,

. F = 108, and

. P = 0.0003.

As is seen, although F is increased substantially and p is reduced, there is still one insignificant term. So, the last run is made with only the $Sin^2(Psi)$ term to give:

. $R^2 = 0.9739$,

. F = 187, and

. P = 0.0000.

---

[8] This measurement was performed on a 304 Stainless steel specimen.

Therefore we can see that the F-value keeps increasing in these three regressions. The best model obtained (that is, the highest F value and lowest p-value) is thus obtained in the last run. The statistical package thus concludes that the model with only the $Sin^2(Psi)$ term is the most appropriate. This example, which has been interpreted as Case I by Dr. Hendricks, is thus also seen as Case I with our statistical procedure.

5.4.1.3 Example of Case II

Experiment 622[9] represents an example of a measurement which has been interpreted by the expert as Case II data. CoStat runs through the three different regressions with the following results (see Figures [5-7], page 72 and [5-8], page 73):

File 622	R^2	F test	p value
Full Model	0.9983	606	0.0001
$Sin^2(Psi)$ Sin(2Psi)	0.9982	1121	0.0000
$Sin^2(Psi)$	0.7541	15	0.0112

Thus we can see that our statistical procedure concludes that the experiment is a Case II (highest F value and lowest P value obtained). It thus matches once again the opinion of our expert.

[9] This experiment was done on a Railroad steel.

5.4.1.4 Example of Case III

This experiment[10] has been classified by our expert as case III data. CoStat runs through the three different regressions with the following results (see Figures [5-9], page 74 and [5-10], page 75):

File 1	R^2	F test	P value
Full Model	0.6265	2.80	0.1485
Sin^2(Psi) Sin(2Psi)	0.6084	4.66	0.0601
Sin^2(Psi)	0.5331	7.99	0.0255

Thus our statistical analysis concludes that this experiment should be seen as a Case I. That gives an example of mis-classification by our procedure. However, we note that in this case both the R^2 and F-values are significantly smaller than was the case for the data described in the previous sections. We found consistently that class 3 data were statistically analyzed as class III[11] or class 1 but with low R^2 and F-values.

---

10 The sample measured was a 410 Stainless Steel.

11 Which is the correct classification since no theoritical model exist for this case. The multiple regression reject Case I, II or IV and thus conclude Case III.

5.4.1.5 Example of Case IV

Experiment 1214[12] represents an example of measurement which has been interpreted as Case IV data. CoStat runs through the three different regressions with the following results (see Figures [5-11], page 76 and [5-12], page 77):

File 1214	R^2	F test	P value
Full Model	0.6878	3.67	0.0977
Sin(2Psi) Tan(Psi)	0.6820	6.43	0.0322
Tan(Psi))	0.2845	2.78	0.1392

This example is also very interesting. Indeed, we can see that our procedure concludes that the best regression is obtained with a model containing only the Sin(2Psi) and Tan(Psi) terms. That is, it classifies experiment 1214 as a case which have not been previously defined! However, again we note that the R^2 and values are not large, thus indicating not a strong fit to the model.

5.4.1.6 Conclusions

We have followed the above procedures several dozen times. For each measurement we have asked our expert to classify the data and then have run three different regressions with CoStat. The results of these analyzes are presented in Figure [5-13], page 78. From these data we see that our statistical analysis, although not 100 % accurate,

12 This measurement was performed on a 304 Stainless steel.

agrees reasonably well with the expert classification. At this point we should remember that this procedure utilizes only a very small amount of the information which is at our disposal. Indeed, we have based our regression only on the d-spacing versus $Sin^2(Psi)$ columns data. It is clear that if we also consider additional information such as the weighted fluctuations in the integrated intensity and perhaps the variations in the FWHM of the diffraction peak, we should be able to discriminate among the classification models. However, the incorporation of such information is not possible in the linear regression as discussed in the previous sections. Also we may have to incorporate in our decision process information concerning the specific material which is under consideration. Indeed, it is well known that some materials (such as austenitic stainless steel, brass or rolled aluminum) have a greater tendency for preferred orientations, for example.

5.4.2 Statistical Pattern Recognition

5.4.2.1 Introduction

All the data collected in our database (see above) have been compiled in a large spreadsheet to perform a cluster analysis[13]. In addition to the regression data, we computed several variables which represent fluctuations in experimental parameters. Among these are the ratios:

- Average/standard deviation of d spacing,
- Average/standard deviation of intensity,
- Average/standard deviation of FWHM, and
- Average/standard deviation of standard deviation

[13] See [Johson, 1988].

We then sorted the data based on variation in these values. With the exception of the integrated intensity, no clear cluster could be found by visual analysis. In this case, it was generally found that large fluctuations in integrated intensity were associated with Class III data. It would appear thus that class III data can potentially be associated with the simultaneous observation of low R^2 and F-test fits to the qualitative models and the simultaneous appearance of large fluctuations in integrated intensities.

Failure to discern clear relationships and/or associations between the fluctuations in the data and given classes of curves is not surprising. It is anticipated that the classification could depend on complex linear (or even non-linear) combinations of these data. Thus, it was decided to find such associations with statistical pattern recognition software. The package we chose is pLogic™ developed by pLogic Knowledge Systems. This statistical method relies on algorithms such as the k-nearest-neighbor rule (k-NN) algorithm[14].

The classification of data with this software consists of two different steps:

- First, one performs a training operation in which the software is fed with manually analyzed data. This corresponds to the learning phase, during which pLogic computes a number of decision rules referred to as "classifiers".

- Then, these classifiers are used by the statistical inference engine of the software to

[14] See [pLogic, 1991] for further details.

provide decisions for the classification of as yet unclassified data.

To follow this procedure and check the accuracy of its decision rules we will use a very common practice in statistical analysis. We will train the system with half of the data manually classified by the expert and then check its accuracy with the other half.

5.4.2.2 Data Training

We randomly divided our measurements into two parts, that is, for each manually analyzed case (I, II, III or IV) we have taken half of the measurements to feed to pLogic. Indeed as explained by Schulman[15]:

> "Any variable selection routine is using the sample to help determine the model. Thus it is entirely possible that we will 'overfit' and tailor the model too precisely to the data at hand. In this event the model will look great for our sample but may not work at all in the population. To protect against this possibility, cross-validation should be used at any time a variable selection routine is employed."

The following fields have been used for each specific measurement:

- Class: this field has been manually entered according to Dr. Hendricks' expert analysis.

[15] See [Schulman, 1991]. The procedure described is for multiple linear regression, but we believe that the same precautions should be used in the case of statistical pattern recognition.

- Filename, Description 1, Description 2, and Material Id Number: these four fields have not been used in the training of the data. They are used only as identifiers.

- Average of (intensity, FWHM, two theta, d-spacing, and standard deviation) as well as the standard deviations for the same fields. These 10 fields completed with the first one (the class type) really constitute the support of the clustering algorithm.

All these data can then be transformed into the correct format required by pLogic[16] and the training phase of the program can be run.

5.4.2.3 Classification of data

The second half of the data can then be put into exactly the same format, except that this time the class type field would be left blank. Indeed, the goal is now for the software to be able to determine the value for this class based on the 10 existing other fields (the five averages and the five standard deviations). The derived classification can then be compared with the expert classification.

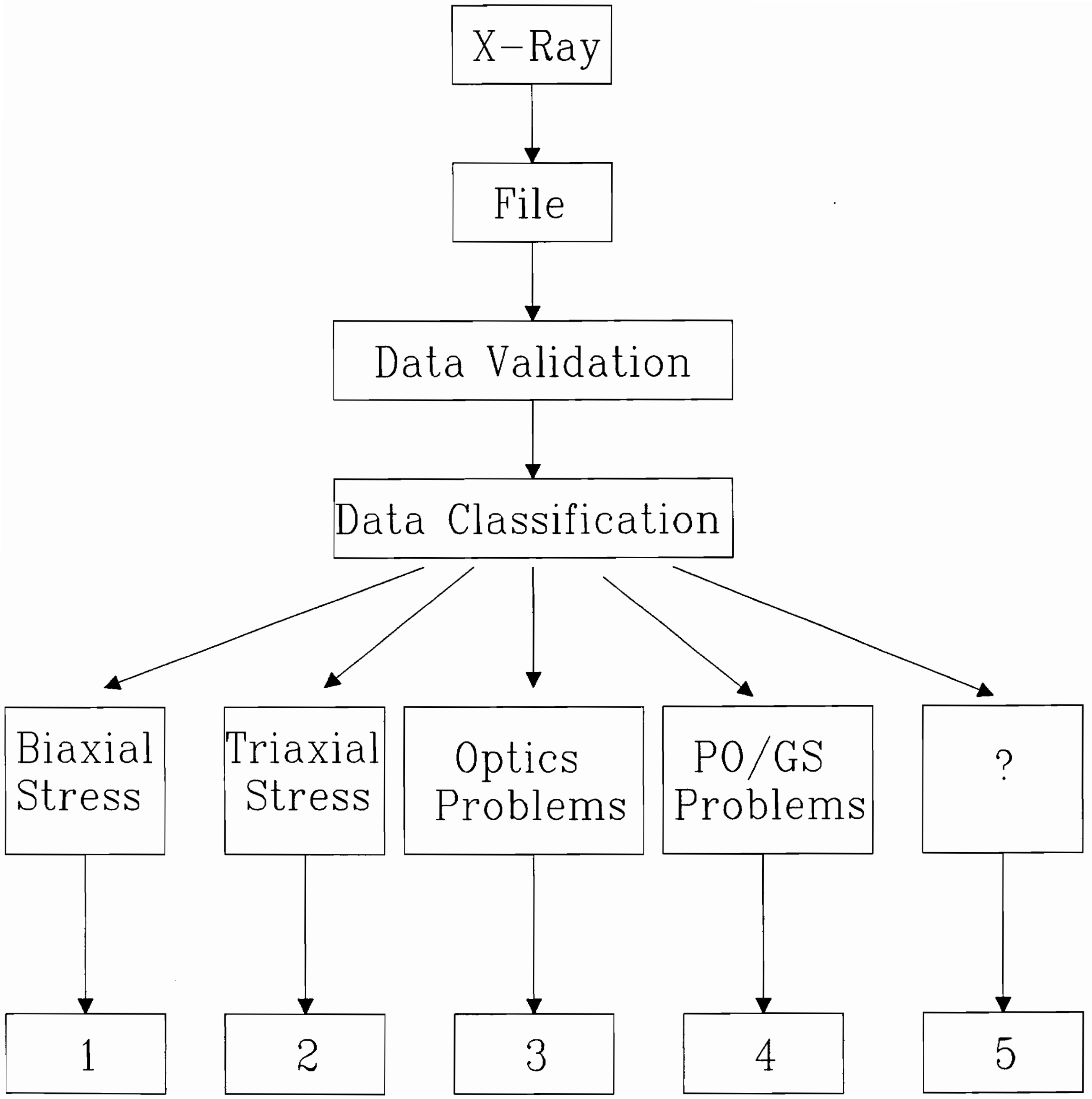
This procedure has not been fully tested in this research; we have simply checked on small sets of data the different features announced. Indeed, we have first chosen to focus on "conventional" statistical analyzes.

---

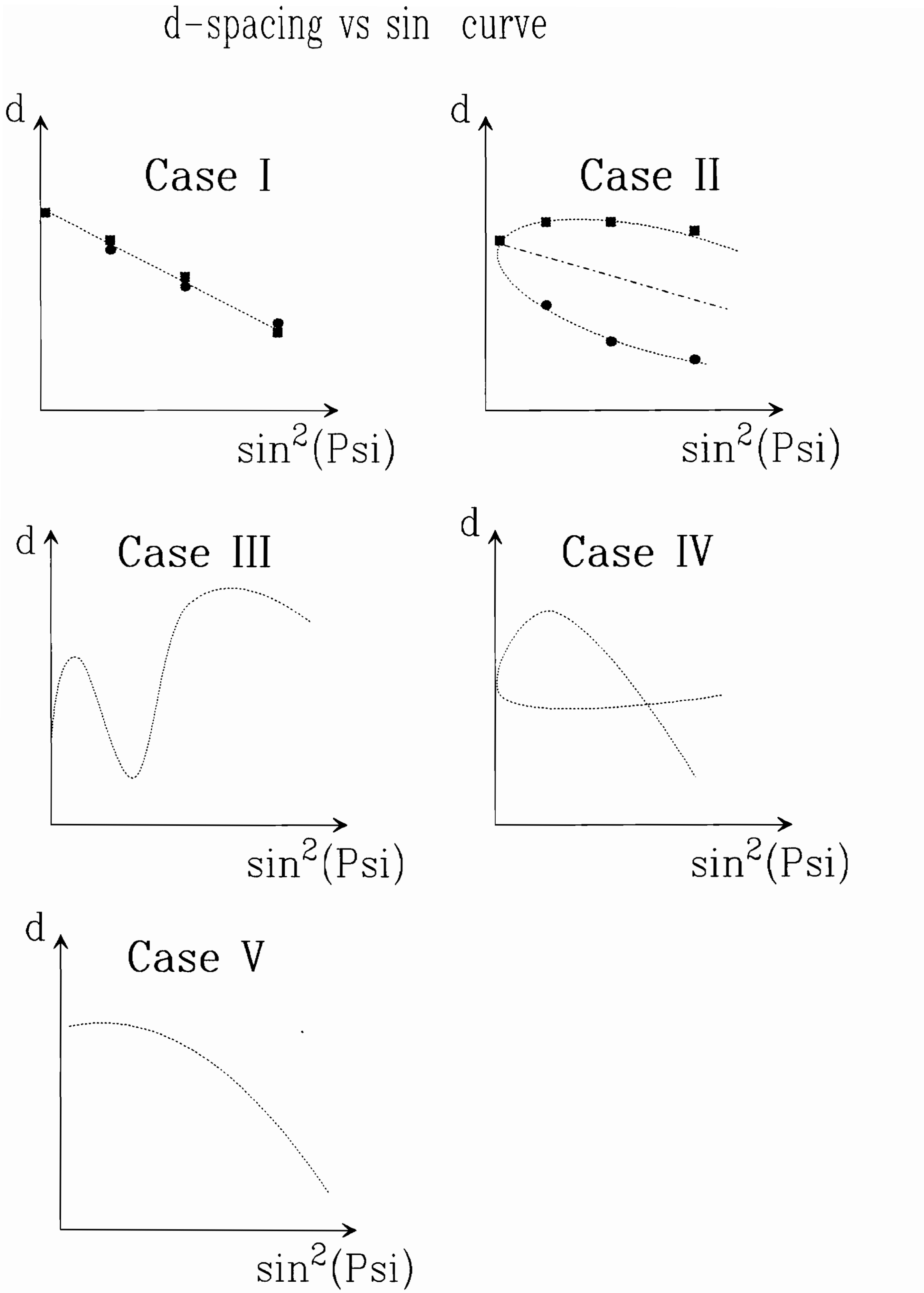
16 Comma separated variables in our case, but the software is also able to recognize dBase format, as well as binary and Lotus 1-2-3 formats.

5.5 Conclusions

We have seen in this chapter that even though we have not yet found a "bullet-proof" procedure to classify our data, a conventional statistical analysis using a "backward multiple regression" shows encouraging results. When coupled with pLogic and when additional data are incorporated, we believe that highly accurate data classification will be possible.



STATISTICAL ANALYSIS - Figure [5-1] Analysis Process



STATISTICAL ANALYSIS - Figure [5-2] d-vs-$\sin^2$Psi curves

Residual Stress Analysis Report

Date : 5-Apr-90 Time : 0:15:42

Sample Description :
Sample No 3 point x=3.5 cycle A side 4
longitudinal stress

System Hardware Configuration :
Auto Psi Angle Drive
Psi Angle Position Encoder

ADC Channels Full Scale		256
Collimator Slit Type	Rectangular	2.00
X-Ray Target Material and Wavelength	Manganese	2.10306
Detector Monting Block Bragg Angle		149.00
PSI Oscillation Angle Range		3.06
High Voltage and Beam Current	35000	1.00
Peak Bounding Range (percent)		20
Material ID Number		80
Material Type	04 Stainless	(Cr 220)
Stress Spectra File Specifications		000604.SPC
Stress Spectra Acquisition Date :	8-Apr-90	0:14:04
Stress Spectra Count Time (Sec)		60
		F10000.149
Calibration File Specifications		

Detector Calibration Coefficients
A 6.628880E-08 B -1.979250E-05 C 6.137930E-02 D 141.1817

Psi	Sin^2(Psi)	Pk Chan	Intens	FWHM	Kalp Cor	2-Theta	D-Spacing	St. Dev
-40.0	0.40314	110.80	211.6	1.13	0.00000	147.83	1.094374	0.000038
-20.0	0.10940	107.43	411.4	1.24	0.00000	147.63	1.094927	0.000031
-10.0	0.02590	105.61	287.4	1.13	0.00000	147.52	1.095228	0.000033
0.0	0.00017	105.21	230.3	1.18	0.00000	147.50	1.095294	0.000033
10.0	0.03469	105.78	290.0	1.17	0.00000	147.53	1.095200	0.000033
20.0	0.12461	107.92	281.8	1.22	0.00000	147.66	1.094847	0.000028
40.0	0.42268	111.91	405.1	1.40	0.00000	147.90	1.094193	0.000029

Fitted Delta D. vs sin^2(Psi) Data		
D Spacing Intercept		1.095249E+00
Slope of Fitted Line		-2.394957E-03
Material Stress Constant		4.950000E-08
Residual Stress	-44.2	-304.6
Counting Statistic Error	1.4	9.7
Goodness of Fit Stress Error	3.5	24.4
Total Stress Error	3.8	26.2

STATISTICAL ANALYSIS - Figure [5-3] Experiment "604"

PRINT DATA
May 14, 1991 12:03:33 pm
Using: C:\SOFT\COHORT\DATA\604.DT
Variable #1: Sin^2(Psi)
Variable #2: Sin(2Psi)
Variable #3: Tan(Psi)
Variable #4: D-Spacing
Replicates (1 - 7)

Rep	Sin^2(Psi)	Sin(2Psi)	Tan(Psi)	D-Spacing
1	0.40314	-0.98106	-0.82185	1.094374
2	0.1094	-0.62428	-0.35048	1.094927
3	0.0259	-0.31767	-0.16306	1.095228
4	1.7E-04	0	0	1.095294
5	0.03469	0.365987	0.18957	1.0952
6	0.12461	0.660552	0.37729	1.094847
7	0.42268	0.987971	0.855653	1.094193

STATISTICAL ANALYSIS - Figure [5-4] New columns for experiment 604.

BACKWARDS MULTIPLE REGRESSION
Apr 22, 1991   9:19:31 pm
Using: C:\SOFT\COHORT2\DATA\604.DT
Regression equation:
D-Spacing = 1.095248366
  + -0.0023809681*Sin^2(Psi)
  + 4.368555E-05*Sin(2Psi)
  + -1.316581E-04*Tan(Psi)

R^2 = 0.9828042843

Source	SS	df	MS	F	P	
Total	1.124859E-06	6				
Regression	1.105516E-06	3	3.685054E-07	57.1540185	.0038	**
Sin^2(Psi)	1.095529E-06	1	1.095529E-06	169.91313613	.0010	***
Sin(2Psi)	8.932564E-09	1	8.932564E-09	1.3854124723	.3241	ns
Tan(Psi)	1.054225E-09	1	1.054225E-09	0.1635068971	.7130	ns
Error	1.934275E-08	3	6.447585E-09			

X	Y observed	Y expected	Residual
1	1.094374	1.0943538476	2.015238E-05
2	1.094927	1.0950067597	-7.975965E-05
3	1.095228	1.0951942896	3.371045E-05
4	1.095294	1.0952479613	4.603872E-05
5	1.0952	1.0951568002	4.319983E-05
6	1.094847	1.0949308569	-8.385689E-05
7	1.094193	1.0941724848	2.051517E-05

Delete Tan(Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
Apr 22, 1991   9:40:51 pm
Using: C:\SOFT\COHORT2\DATA\604.DT
Regression equation:
D-Spacing = 1.0952489017
  + -0.0023865238*Sin^2(Psi)
  + -5.461431E-05*Sin(2Psi)

R^2 = 0.9818670783

Source	SS	df	MS	F	P	
Total	1.124859E-06	6				
Regression	1.104462E-06	2	5.522309E-07	108.29661776	.0003	***
Sin^2(Psi)	1.095529E-06	1	1.095529E-06	214.84149289	.0001	***
Sin(2Psi)	8.932564E-09	1	8.932564E-09	1.7517426291	.2562	ns
Error	2.039698E-08	4	5.099245E-09			

STATISTICAL ANALYSIS - Figure [5-5] F-Test 604 (First Part)

X	Y observed	Y expected	Residual
1	1.094374	1.0943403784	3.362157E-05
2	1.094927	1.0950219106	-9.491062E-05
3	1.095228	1.0952044401	2.355994E-05
4	1.095294	1.095248496	4.550401E-05
5	1.0952	1.0951461251	5.387494E-05
6	1.094847	1.0949154414	-6.844138E-05
7	1.094193	1.0941862085	6.791522E-06

Delete Sin(2Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
Apr 22, 1991 9:41:02 pm
Using: C:\SOFT\COHORT2\DATA\604.DT
Regression equation:
D-Spacing = 1.0952493736
+ -0.002393931*Sin^2(Psi)

R^2 = 0.973926025

Source	SS	df	MS	F	P	
Total	1.124859E-06	6				
Regression	1.095529E-06	1	1.095529E-06	186.76208824	.0000	***
Sin^2(Psi)	1.095529E-06	1	1.095529E-06	186.76208824	.0000	***
Error	2.932954E-08	5	5.865908E-09			

X	Y observed	Y expected	Residual
1	1.094374	1.0942842842	8.971575E-05
2	1.094927	1.0949874775	-6.047754E-05
3	1.095228	1.0951873708	4.062922E-05
4	1.095294	1.0952489666	4.503338E-05
5	1.0952	1.0951663281	3.367188E-05
6	1.094847	1.0949510658	-1.040658E-04
7	1.094193	1.0942375068	-4.450684E-05

BACKWARDS MULTIPLE REGRESSION
Apr 28, 1991 8:36:56 pm
Using: C:\SOFT\COHORT2\DATA\622.DT
Regression equation:
D-Spacing = 1.1727918144
+ -0.002390514*Sin^2(Psi)
+ -3.779407E-04*Sin(2Psi)
+ 4.319721E-05*Tan(Psi)

R^2 = 0.9983514547

Source	SS	df	MS	F	P	
Total	1.92123E-06	6				
Regression	1.918062E-06	3	6.393542E-07	605.59588146	.0001	***
Sin^2(Psi)	1.448817E-06	1	1.448817E-06	1372.3185091	.0000	***
Sin(2Psi)	4.689913E-07	1	4.689913E-07	444.22831805	.0002	***
Tan(Psi)	2.542413E-10	1	2.542413E-10	0.240817229	.6573	ns
Error	3.167232E-09	3	1.055744E-09			

X	Y observed	Y expected	Residual
1	1.171942	1.1719486962	-6.696183E-06
2	1.172532	1.1725076505	2.43495E-05
3	1.172788	1.1728129854	-2.498538E-05
4	1.172783	1.1727916949	-8.6949E-06
5	1.172479	1.1724440408	3.495919E-05
6	1.17186	1.1718840994	-2.409937E-05
7	1.171265	1.1712598329	5.167149E-06

Delete Tan(Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
Apr 28, 1991 8:36:58 pm
Using: C:\SOFT\COHORT2\DATA\622.DT
Regression equation:
D-Spacing = 1.1727917378
+ -0.0023897501*Sin^2(Psi)
+ -3.427885E-04*Sin(2Psi)

R^2 = 0.9982191221

Source	SS	df	MS	F	P	
Total	1.92123E-06	6				
Regression	1.917808E-06	2	9.589041E-07	1121.0424401	.0000	***
Sin^2(Psi)	1.448817E-06	1	1.448817E-06	1693.7931541	.0000	***
Sin(2Psi)	4.689913E-07	1	4.689913E-07	548.29172601	.0000	***
Error	3.421473E-09	4	8.553683E-10			

STATISTICAL ANALYSIS - Figure [5-7] F-Test 622 (First Part)

X	Y observed	Y expected	Residual
1	1.171942	1.1719564409	-1.444087E-05
2	1.172532	1.1725021186	2.988145E-05
3	1.172788	1.1728070579	-1.905792E-05
4	1.172783	1.1727916183	-8.618306E-06
5	1.172479	1.1724500998	2.890019E-05
6	1.17186	1.1718896452	-2.964517E-05
7	1.171265	1.1712520194	1.298063E-05

Delete Sin(2Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
Apr 28, 1991 8:37:00 pm
Using: C:\SOFT\COHORT2\DATA\622.DT
Regression equation:
D-Spacing = 1.1727949182
+ -0.002410087*Sin^2(Psi)

R^2 = 0.7541091465

Source	SS	df	MS	F	P
Total	1.92123E-06	6			
Regression	1.448817E-06	1	1.448817E-06	15.334225323	.0112 *
Sin^2(Psi)	1.448817E-06	1	1.448817E-06	15.334225323	.0112 *
Error	4.724128E-07	5	9.448256E-08		

X	Y observed	Y expected	Residual
1	1.171942	1.1716068417	3.351583E-04
2	1.172532	1.172205676	3.26324E-04
3	1.172788	1.1726416367	1.463633E-04
4	1.172783	1.1727947977	-1.179769E-05
5	1.172479	1.1726265977	-1.475977E-04
6	1.17186	1.1721856482	-3.256482E-04
7	1.171265	1.171587802	-3.22802E-04

STATISTICAL ANALYSIS - Figure [5-8] F-Test 622 (Second Part)

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:16:44 pm
Using: C:\SOFT\COHORT\DATA\000001.DT
Regression equation:
D-Spacing = 1.1761681021
  + 8.132239E-04*Sin^2(Psi)
  + 0.0015300865*Sin(2Psi)
  + -0.0028439123*Tan(Psi)

R^2 = 0.6264923722

Source	SS	df	MS	F	P	
Total	1.689451E-06	8				
Regression	1.058428E-06	3	3.528094E-07	2.7955358208	.1485	ns
Sin^2(Psi)	9.005674E-07	1	9.005674E-07	7.1357757211	.0443	*
Sin(2Psi)	1.272955E-07	1	1.272955E-07	1.0086438629	.3613	ns
Tan(Psi)	3.056521E-08	1	3.056521E-08	0.2421878783	.6435	ns
Error	6.310228E-07	5	1.262046E-07			

X	Y observed	Y expected	Residual
1	1.176291	1.1761684274	1.225726E-04
2	1.175661	1.1762181407	-5.571407E-04
3	1.17669	1.176224995	4.65005E-04
4	1.176152	1.176203767	-5.176697E-05
5	1.176281	1.1761404896	1.405104E-04
6	1.176086	1.1760228805	6.31195E-05
7	1.175669	1.1758372697	-1.682697E-04
8	1.175425	1.1755634365	-1.384365E-04
9	1.175303	1.1751785936	1.244064E-04

Delete Tan(Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:16:46 pm
Using: C:\SOFT\COHORT\DATA\000001.DT
Regression equation:
D-Spacing = 1.1761054803
  + -0.0031560897*Sin^2(Psi)
  + 7.92517E-04*Sin(2Psi)

R^2 = 0.6084005682

Source	SS	df	MS	F	P	
Total	1.689451E-06	8				
Regression	1.027863E-06	2	5.139314E-07	4.6608895597	.0601	ns
Sin^2(Psi)	9.005674E-07	1	9.005674E-07	8.1673254222	.0289	*
Sin(2Psi)	1.272955E-07	1	1.272955E-07	1.1544536972	.3239	ns
Error	6.61588E-07	6	1.102647E-07			

STATISTICAL ANALYSIS - Figure [5-9] F-Test 1 (First Part)

X	Y observed	Y expected	Residual
1	1.176291	1.1761042178	1.867822E-04
2	1.175661	1.1762875333	-6.265333E-04
3	1.17669	1.1762842871	4.057129E-04
4	1.176152	1.1762285095	-7.650948E-05
5	1.176281	1.1761196392	1.613608E-04
6	1.176086	1.1759643132	1.216868E-04
7	1.175669	1.1757680706	-9.907062E-05
8	1.175425	1.1755336308	-1.086308E-04
9	1.175303	1.1752677985	3.520149E-05

Delete Sin(2Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:16:48 pm
Using: C:\SOFT\COHORT\DATA\000001.DT
Regression equation:
D-Spacing = 1.1763634918
 + -0.001858711*Sin^2(Psi)

R^2 = 0.5330533329

Source	SS	df	MS	F	P	
Total	1.689451E-06	8				
Regression	9.005674E-07	1	9.005674E-07	7.9910053758	.0255	*
Sin^2(Psi)	9.005674E-07	1	9.005674E-07	7.9910053758	.0255	*
Error	7.888835E-07	7	1.126976E-07			

X	Y observed	Y expected	Residual
1	1.176291	1.1763627483	-7.174828E-05
2	1.175661	1.1762956302	-6.346302E-04
3	1.17669	1.1762182893	4.717107E-04
4	1.176152	1.1761222125	2.978752E-05
5	1.176281	1.1760025858	2.784142E-04
6	1.176086	1.1758674575	2.185425E-04
7	1.175669	1.1757208052	-5.180524E-05
8	1.175425	1.1755645992	-1.395992E-04
9	1.175303	1.175403672	-1.00672E-04

STATISTICAL ANALYSIS - Figure [5-10] F-Test 1 (Second Part)

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:15:27 pm
Using: C:\SOFT\COHORT\DATA\1214.DT
Regression equation:
D-Spacing = 1.1700629138
+ -1.437497E-04*Sin^2(Psi)
+ 0.002204858*Sin(2Psi)
+ -0.0038227056*Tan(Psi)

R^2 = 0.6878467345

Source	SS	df	MS	F	P	
Total	2.851749E-07	8				
Regression	1.961566E-07	3	6.538554E-08	3.672590795	.0977	ns
Sin^2(Psi)	1.752733E-09	1	1.752733E-09	0.0984479468	.7664	ns
Sin(2Psi)	6.339559E-08	1	6.339559E-08	3.5608189132	.1178	ns
Tan(Psi)	1.310083E-07	1	1.310083E-07	7.358505525	.0421	*
Error	8.901827E-08	5	1.780365E-08			

X	Y observed	Y expected	Residual
1	1.170298	1.1703227832	-2.478317E-05
2	1.170117	1.1700199057	9.709434E-05
3	1.169933	1.1699785378	-4.553775E-05
4	1.169868	1.1700133332	-1.453332E-04
5	1.170034	1.1700629138	-2.891378E-05
6	1.170309	1.1701105523	1.984477E-04
7	1.17017	1.1701387008	3.129916E-05
8	1.169956	1.1700719689	-1.159689E-04
9	1.169764	1.1697303044	3.36956E-05

Delete Sin^2(Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:15:29 pm
Using: C:\SOFT\COHORT\DATA\1214.DT
Regression equation:
D-Spacing = 1.1700499966
+ 0.0022053422*Sin(2Psi)
+ -0.0038235256*Tan(Psi)

R^2 = 0.6819635292

Source	SS	df	MS	F	P	
Total	2.851749E-07	8				
Regression	1.944789E-07	2	9.723944E-08	6.4328803455	.0322	*
Sin(2Psi)	6.34139E-08	1	6.34139E-08	4.195150292	.0865	ns
Tan(Psi)	1.31065E-07	1	1.31065E-07	8.6706103991	.0258	*
Error	9.069602E-08	6	1.5116E-08			

STATISTICAL ANALYSIS - Figure [5-11] F-Test 1214 (First Part)

X	Y observed	Y expected	Residual
1	1.170298	1.170345778	-4.777799E-05
2	1.170117	1.1700237612	9.323875E-05
3	1.169933	1.1699699336	-3.69336E-05
4	1.169868	1.170001499	-1.33499E-04
5	1.170034	1.1700499966	-1.599661E-05
6	1.170309	1.1700987558	2.102442E-04
7	1.17017	1.1701301589	3.984108E-05
8	1.169956	1.1700758833	-1.198833E-04
9	1.169764	1.1697532334	1.076656E-05

Delete Sin(2Psi) from the model.

BACKWARDS MULTIPLE REGRESSION
May 9, 1991 10:15:31 pm
Using: C:\SOFT\COHORT\DATA\1214.DT
Regression equation:
D-Spacing = 1.1700499454
  + -2.837548E-04*Tan(Psi)

R^2 = 0.2845170252

Source	SS	df	MS	F	P
Total	2.851749E-07	8			
Regression	8.113711E-08	1	8.113711E-08	2.7836010583	.1392 ns
Tan(Psi)	8.113711E-08	1	8.113711E-08	2.7836010583	.1392 ns
Error	2.040378E-07	7	2.914825E-08		

X	Y observed	Y expected	Residual
1	1.170298	1.1702135301	8.446994E-05
2	1.170117	1.1701531187	-3.611866E-05
3	1.169933	1.1700999771	-1.669771E-04
4	1.169868	1.1700748109	-2.068109E-04
5	1.170034	1.1700499454	-1.594542E-05
6	1.170309	1.1700249333	2.840667E-04
7	1.17017	1.1699997954	1.702046E-04
8	1.169956	1.1699466561	9.34388E-06
9	1.169764	1.1698862331	-1.222331E-04

STATISTICAL ANALYSIS - Figure [5-12] F-Test 1214 (Second Part)

	Model		F					
FileName	RWH	Stat	Model	A	B	C	RSquare	MatId
731	1	1	11.5	11.5			0.62	8
265	1	1	12	12			0.63	80
184	1	1	15	15			0.75	140
270	1	1	17	17			0.71	80
116	1	1	24	24			0.77	80
174	1	1	47	47			0.9	140
192	1	1	59	59			0.92	140
193	1	1	85	85			0.94	140
175	1	1	91	91			0.95	140
180	1	1	122	122			0.96	140
795	1	1	123	123			0.96	80
800	1	1	126	126			0.96	80
604	1	1	187	187			0.97	80
179	1	1	360	360			0.986	140
713	1	1	380	380			0.987	80
211	1	2	18	19	17		0.86	80
119	1	2	33	51	14		0.92	80
224	1	2	38	56	20		0.93	80
187	1	2	86	163	11		0.97	142
194	1	2	1124	2240	8.4		0.998	140
135	1	?					0.25	80
39	1	?					0.18	108
195	1	?					0.3	140
287	1	?					0.36	80
846	1	?					0.43	80
1149	2	1	18.49	18.49			0.73	81
612	2	1	85	85			0.95	80
853	2	2	8.3	11.2	5.2		0.8	57
851	2	2	18.5	25.5	11.5		0.9	57
858	2	2	38	52	24		0.95	57
839	2	2	46	70	23		0.96	57
838	2	2	71	119	23		0.97	57
622	2	2	1100	1700	550		0.998	57
398	2	4	11.6	0.1	9.7	25	0.87	80
848	2	4	23	37	27	5	0.96	57
849	2	4	89	193	72	3.8	0.989	57
897	2	4	234	555.6	134	13.3	0.996	57
863	2	4	241	659	51	12	0.996	57
1148	2	*	42		76.6	7.41	0.93	81
40	2	?					0.16	108
845	2	?					0.15	57
856	2	?					0.56	57
14	3	1	6.54	6.54			0.48	80
729	3	1	7.23	7.23			0.51	8
1	3	1	8	8			0.53	83
189	3	1	9.4	9.4			0.65	140

STATISTICAL ANALYSIS - Figure [5-13] Compilation (Part 1)

316	3	1	15	15			0.65	80
182	3	1	19.34	19.34			0.79	140
185	3	1	60	60			0.92	140
183	3	1	82	82			0.94	140
739	3	2	5.8	5	6.8		0.66	8
745	3	2	7.6	8.96	6.2		0.72	8
728	3	2	31.6	48	15.1		0.91	8
746	3	*	10.5		10.5		0.61	8
747	3	*	24		24		0.78	8
38	3	?					0.19	108
730	3	?					0.33	8
732	3	?					0.32	8
733	3	?					0.31	8
734	3	?					0.12	8
736	3	?					0.35	8
737	3	?					0.3	8
740	3	?					0.23	8
741	3	?					0.29	8
813	3	?					0.19	80
203	4	1	6.8	6.8			0.58	7
738	4	1	15.6	15.6			0.69	8
301	4	1	61	61			0.9	112
21	4	2	81	133	28		0.98	7
1213	4	4	8.2	3.9	15.6	5.2	0.83	81
870	4	4	11.8	11.5	0.44	23.4	0.92	57
201	4	4	26	36	15	27	0.96	7
1214	4	?					0.28	81
409	4	?					0.05	80
22	4	?					0.18	7
37	4	?					0.18	156
42	4	?					0.16	108
186	4	?						140
200	4	?					0.41	7

STATISTICAL ANALYSIS - Figure [5-14] Compilation (Part 2)

6. SYSTEM INTEGRATION

If every instrument could accomplish its own work, obeying or anticipating the will of others..., if the shuttle could weave, and the pick touch the lyre, without a hand to guide them, chief workmen would not need servants, nor masters slaves.
Aristotle.

Everything should be made as simple as possible, but no simpler.
Albert Einstein

This chapter discusses the integration of the various components of the system described in the three previous chapters. These are: the importation of data from Excel into Nexpert Object; the use of external procedural language data analysis routines; the use of external graphics routines; the database management system; the user interface and the callable interface. Each is discussed in the following sections.

6.1 Data Transfer

We have developed[1] two programs to transfer data generated on the TEC Model 1600 series of Portable X-ray Stress Analysis Systems to a database management system residing on an IBM PC or near compatible computer. The programs have been developed without need for any programming of the DEC LSI-11 computer or its RT-11 operating system. Rather, we have developed a program which captures the ASCII characters transferred to the Epson MX-80 printer attached to the LSI-11 computer and which presents the user with a carbon copy of the report on the personal computer. The second program strips all of the unnecessary textual information from the captured printout and converts it to a Sylk file suitable for importation into a number of commercial spreadsheets such as Microsoft Excel.

6.1.1 Introduction

Technology for Energy Corporation (TEC) has developed a series of X-ray stress analysis systems designed for portable use in the field. Several other manufacturers have developed similar portable systems, or have developed more traditional laboratory systems designed for measurements on smaller samples. For the most part, these systems have been developed using different brands of computers for on-line data acquisition and control. For our expert system, it is required that data from these various systems be available on the same (independent) computer. In other laboratories, several machines from different vendors

[1] The programs described in this section have been written by Jac Potet in partial fulfillment of his Projet de Fin d'Etudes, both in ENSAM, Paris and VPI&SU. See [Potet, 1991].

co-exist side by side. In other cases, the users need to archive and/or analyze their data beyond the capability provided by the manufacturers[2].

There are several ways in which this problem can be resolved. These are outlined schematically in Figure [6-1], page 101. First, the data provided by the vendor in his printed stress reports can be laboriously manually retyped into the new system as was done early in our development. This is hardly a viable solution in the computer age, especially when tens of thousands of data sets may be involved. If the stress analysis computer is equipped with a good-quality printer (e.g. an ink jet or perhaps a laser printer), it is possible to transfer the data to the second (different) computer through a laser scanner and optical character recognition software. However, as we have discovered, this process is not reliable when the print quality is poor to marginal, as is usually the case with most dot matrix printers. Of course, the most obvious solution is to write a small program for the stress analyzer host computer which reads the manufacturers file, reformats the data in the desired manner, and outputs the data, as an ASCII file, on an RS232C asynchronous port for pickup by the second computer. This method, the most intellectually appealing of all, suffers some limitations depending on the computer installed in the stress analyzer. In several instances, the vendor's file structures are unavailable[3], or the cost of a language compiler for that machine is prohibitive[4]. Thus, for many reasons, a different and more

---

[2] For example, most vendors do not include triaxial stress analysis with the software provided with their instruments.

[3] This information is often considered proprietary.

[4] For example, the Fortran compiler for DEC's LSI-11 series

reasonable option must be found. It is highly desirable that such an alternative be independent of both the hardware and software of the host stress analysis computer in order that the user can easily develop, implement, maintain, and modify the link to suit his changing needs.

6.1.2 Hardware requirements

The software described in the next Section has been designed to operate with the TEC Model 1600 series X-ray stress analysis systems which are based on a DEC LSI-11 microcomputer system with printer output directed to an Epson or Epson compatible printer connected to an RS232C asynchronous serial port. The software will not operate on early systems which were connected to the parallel port. Furthermore, the software has been written to operate on an IBM or near compatible personal computer with input also being received on an asynchronous RS232C serial port. In order to control the interaction between the two computers, it is necessary to insert a null modem cable between the two machines. Such a cable is available from most computer stores and/or computer supply vendors. Details of this connection can be found in [Potet, 1990].

---

of machines carries a $1500 licensing fee. In addition, current versions of the compiler require that the RT-11 operating system be upgraded at a further cost of $1000. Finally, none of the software is available on the 8-inch disks provided with the TEC instrument. Thus, a further investment of several thousand dollars is required to upgrade the disk drives to 5-1/4 inch technology.

6.1.3 Program descriptions

The process of moving data from the TEC stress analyzer to the PC has been implemented in two steps in order to provide simplicity for reprogramming the conversion when conditions warrant. For example, TEC2ASC can also be used to transfer raw peak intensity data provided by the TEC program SPECTRUM without modification. However, ASC2SLK cannot be used to import these data to the DBMS. However, it would be a trivial matter to develop a new program to convert the PC file to a quote and comma delimited ASCII file which is compatible with almost all spreadsheet and database management programs.

6.1.3.1 TEC2ASC

TEC2ASC is a classical communication program using the serial RS232C connection protocol. It is composed of two separate programs:

- JEPENSE.C written in C language, and
- JESUIS.ASM written in assembly code.

The C program creates an external ASCII file from the buffer stored in memory. This program also asks for the input filename and calls the assembly language modules.

The assembly modules start by setting up the different parameters useful for the communication, that is:

- 9600 bits per second,
- Parity = Odd,
- 7 data bits, and
- 1 stop bit.

Then it starts to scrutinize the RS232C waiting for the first bit to arrive, store all the data in memory and wait for a key to be struck. This program stores all the information received from the TEC machine in a memory buffer. Therefore the file created is an exact copy of the TEC stress analysis report but which now exists as an ASCII file on the PC.

6.1.3.2 ASC2SLK

During the first part of this project this program was in charge of transforming the ASCII output file previously created with TEC2ASC into a Sylk file readable into a spreadsheet such as Excel. First it opens the three files it will need:

- the model file : MODEL.SLK (i.e. the blank superset of data),
- the ASCII file : FIRST.TRY, and
- the Sylk file : FIRST.SLK.

The general concept is to merge MODEL.SLK and FIRST.TRY into FIRST.SLK. In order to do that the programs proceed with the following sequence for all the data fields: first locate the correct field to be computed in FIRST.TRY, then find the location of the same field in MODEL.SLK, copy the corresponding model file part into FIRST.TRY and finally copy the field value from FIRST.TRY into FIRST.SLK. This program also incorporates a special routine to eliminate all the useless blanks read from the source file FIRST.TRY. It has to be noticed that this program has been developed with separate C-routines and thus is easily adaptable to any other kind of conversion (from ASCII to a Lotus 123 WKS file for example).

This procedure has been updated in the second part of our project to incorporate the database management system developed by S. Courtney (See [Courtney, 1990] and [Courtney, 1991]). Indeed, this program takes the data from an ASCII format and feeds it into the DBMS. When the expert system needs to analyze this file, it triggers a C routine which makes the query to the dBase database. Thus we have developed a very general concept, that is the data can come from any X-ray stress analyzer, be fed into the database and from there be accessed by the expert system independently of its experimental origin.

6.1.4 Program user guide

Detailed instructions for the use of the programs TEC2ASC and ASC2SLK are provided in [Potet, 1990].

6.1.5 Using Excel

We have created a template which mirrors the superset stress analyzer printer output for use under EXCEL as shown in Figure [3-3], page 29 and Figure [6-2], page 102. This template is a superset of the data prepared by as many commercial stress analysis systems for which we have been able to obtain information. To load data into the spreadsheet, run Excel and load the template sheet. Then select data import and provide the name of the Sylk file created by ASC2SLK.

It is important to keep in mind that the format of the data created by ASC2SLK is set up specifically for this spreadsheet template. If you wish to use a different template, you must reprogram the program ASC2SLK as described in [Potet, 1990]. Note that with this simple system, one may input residual stress data into a spreadsheet, perform simple graphics and data analysis without the expert system. It is a stand-alone system.

6.2 Database management system

6.2.1 Introduction

A significant problem concerns the management of large amounts of data generated by modern, computer-controlled stress analysis systems. In the case of every commercial system with which we are familiar, all of the data associated with an individual stress measurement are stored on either floppy or hard disk, or are output directly to a printer. When one realizes that such systems often are used to make thousands, or even tens of thousands, of stress measurements per year, the magnitude of the problem associated with managing so much data becomes apparent.

In response, we have begun work[5] on a program that will catalogue measurements generated on commercial analysis systems and which will ultimately allow end users to manipulate large quantities of data by use of simple spreadsheet commands. This program will be able to run both independently and in connection with our expert system. The development of our expert system places certain requirements on the database program:

- it must accept data from any and all commercial residual stress analysis systems, and

[5] See [Courtney, 1991].

- it must operate independently of any particular commercial system.

For reasons previously outlined we have chosen to develop our system on a 386-class IBM-compatible PC. Because many of the commercial stress analysis systems do not utilize this technology for their computer-control system, we developed a method[6] by which data may be transferred from any commercial system to an IBM PC, and then entered into our expert system through a Microsoft Excel spreadsheet. Although successful for small amounts of data, this approach suffers from the fact that each stress measurement point must be retained on the system disk as an individual file. To resolve this problem, and yet remain consistent with the philosophy of the development of the expert system previously described, we have undertaken the development of a true X-ray stress analysis database management system (DBMS) that is written in dBase III language and compiled with Clipper 5.0. In the next section we outline the requirements of an appropriate database management system and the reasons for our selection of the dBase format.

6.2.2 Selection of software

Two primary databases will be linked in the DBMS. The first, the "X-ray database", will record x-ray measurements that are automatically generated by commercial analysis systems and the second, the "materials database", will hold certain materials constants that will be used by the expert system in determining the accuracy of the measurements.

---

[6] See previous section.

Both of the primary databases must be readily accessible from within the expert system. They must be packaged into an integrated program that permits user edits, queries and reports. Furthermore, it is apparent that a DOS-executable append routine will be necessary to transfer commercial analyses into the X-ray database. These functions can be handled adequately by a number of database programs now available on the market. However, we are interested in several non-traditional database operations, such as using the DBMS to perform low-level computations that are not handled efficiently within the expert system. We also would like to permit users the option of manipulating X-ray measurements from a spreadsheet environment, where individual data analysis and preparation of graphs would be greatly facilitated.

Our requirements may be summarized as follows:

- stress measurements data must be entered into the X-ray database automatically by external programs,

- data must be available for use by external programs without invocation of the DBMS or the creation of import/export files, and

- data must be available for simple user interaction in a spreadsheet environment.

In short, all X-ray databases will be stored in dBase format and packaged into a separate, compiled DBMS program, which will be able to serve as a stand alone access and retrieval catalogue for X-ray residual stress measurements. Unprecedented reporting and presentation capabilities could then be obtained through use of a third-party spreadsheet program.

6.2.3 Current system

The DBMS is being written in dBase III language and compiled with Clipper 5.0. It will be offered to researchers as a separate means of efficiently storing residual stress measurements. It will be used by the expert system as a source of raw data, as an interface with several commercial analysis systems, and as an efficient calculator. To understand the nature of its interaction with our expert system, some mention of existing expert system data access protocol is necessary.

Current transfer of X-ray measurements is accomplished as follows: the X-ray analysis is redirected from printer to file, where it is converted to Sylk format as previously described. The Sylk file subsequently is incorporated into a flat file database in Microsoft Excel (the user runs Excel, then imports data into a previously defined outline). The expert system is able to access this data directly. This approach has several drawbacks: the Sylk file is unwieldy, separate files are required with each X-ray analysis, the user must own a copy of Microsoft Excel, and the process lacks automation.

The DBMS is considerably more straightforward: it contains a C link that appends each X-ray analysis in the form of delimited text into a dBase database. If the residual stress measurement system is remote from the expert system or central data repository, a temporary database may be used, with data from this repository periodically updated to the main X-ray database kept on the expert system computer. A database driver, written in C, enables the expert system to read and modify the X-ray and materials data.

Rather than re-creating exhaustive materials databases, such as the American Society for Metals' Mat.DB, our materials database holds a minimal amount of information needed by the expert system for only those materials currently supported by X-ray analysis vendors. Through the database, users have the option of adding materials that appear in new product releases. Because vendors typically provide residual stress information for the more common engineering materials (for instance, 304 stainless steel), many of the materials in this database will overlap.

We recognize that vendors include proprietary materials codes and constants into their commercial systems. Therefore, we will prepare a separate vendor database that will contain information specific to their commercial systems[7]. Users then will simply choose their X-ray analysis equipment from a list provided within the DBMS; this one-time entry will automatically correlate appropriate materials with codes generated from on-site equipment.

Naturally, in order to develop this type of vendor database, we must receive a list of materials used in each of the major commercial systems. Currently, we have specific materials data from only two of the 12 or more companies that prepare X-ray residual stress analyzers.

---

[7] We also note that this file can be encrypted and thus kept confidential.

6.2.4 Development

The database management program has been developed in two stages:

(1) creation of the X-ray and materials database structures, and

(2) programming of the append routine and the drivers to allow the expert system to access data directly without user interface.

Data input into both the X-ray and materials databases will continue through the length of the project in order to assist in development of the DBMS and expert system. However, because it is superfluous, X-ray data generated in Virginia Tech's laboratories will not be included with the completed DBMS.

6.3 Importation of data from Excel

6.3.1 Excel superset

Based on the input from several experts worldwide (see Section 3.3.1, page 22), we have developed a list of the variables required to completely describe an arbitrary residual stress measurement in any material. This list of variables is a "superset" of the variables commonly encountered in all commercial X-ray stress analysis systems.

To import data from the TEC machine (and with the desire to be able to import data from any other X-ray machine) this "superset" has been implemented using Excel. It consists of a worksheet including all the data produced by the X-ray machine as illustrated in Figure [6-2], page 102.

During the first part of this project, data from the TEC machine currently operated in Prof. Hendricks' laboratory has been imported by hand. Then an electronic link was programmed. Indeed an instrument-specific data driver (see step 5 in Figure [3-1], page 27) has been written by Jac Potet using C and assembly language. Furthermore, the system has been designed so that other data translation drivers may be implemented easily for any diffractometer available on the market. The only requirement is to have a data file available in ASCII format.

It is an absolutely normal spreadsheet from the Excel point of view, except that it must be transformed to a Sylk format in order to be read by Nexpert Object. This is accomplished using the conventional export commands of Excel.

The spreadsheet can be easily updated to accommodate the features of the different X-ray machines to contain all of their data. All future machine-dependent drivers which must be created to avoid hand-importation will only have to consider the well known and recognized format of Excel. It should even be possible to quickly develop such drivers which can be triggered using only Excel macros.

6.3.2 Static data retrieval

The first method of data retrieval was to import the "normal" TEC output into Nexpert Object. Indeed, in this case the general design of the worksheet is "static". This means that all of the cells to be used are identified and known (but not their contents, which are obviously dependent on the specific measurement). The only difference which can exist between two different sheets is related to the number of psi-angles chosen by the operator (from 5 to 10 on the TEC machine).

In this particular retrieval scheme all the data cells (shadowed in grey in Figure [6-2], page 102) have been given a name. Thus for example:

. cell **D5** has the name "Measurement.Type", and

. cell **H14** has the name "Material.Name".

These names precisely match an object name previously defined in Nexpert Object. For this reason this method is called "static" as it is necessary to know what data are going to be read into Excel when Nexpert Object is programmed.

During the interaction with Nexpert Object the operator is asked the name of the specific sheet which contains the data to be analyzed and then the importation of all the data in the different cells of the spreadsheet automatically fills all the different objects created in Nexpert Object. Thus, the different stages of the expertise (validation and interpretation of the data) are able to start.

6.3.3 Dynamic data retrieval

The previous method has been used to import the well known (and fixed) outputs of the X-ray analysis machine. However, to retrieve the materials-specific data from the materials database the problem becomes more complex. Indeed, since we want to allow the user to update this database (and to do so he will only have to know how to use a simple spreadsheet) we can expect him or her to fill it with information not previously known by Nexpert Object. Thus, the previous method ("static retrieval") in which the Excel cells exactly match the Nexpert Object property names cannot be used. For this reason a dynamic retrieval of data has been chosen[8]. This retrieval must be selective. In other words, we want to import only the data relative to the material for which we are interpreting the measurements. Indeed, we can expect the material knowledge base to grow with all the materials added by the users, and thus we could face memory problems in the end-user computer. Therefore, when the first importation of the X-ray output is being made Nexpert Object can have access to the Material Id Number in cell number E14 (see Figure [6-2], page 102) and retrieve the material data related to it. To do this, Nexpert Object first must create a dynamic object with the name :

"Material_" + Material_Id_Number.

Thus, for example, the objects "Material_83" or "Material_199" can be created. This dynamic object is dynamically linked to the "materials" class which contains the properties :

. Designation: brief description of the material,

. I_D: Identification number as in the TEC or other system manual,

---

[8] During the static retrieval of information all the spreadsheet cells have been given a name which is known in Nexpert Object. On the other hand to retrieve information about a new material added by the user we need such a dynamic retrieval. Indeed, in this latter case the name defined by the user cannot be known in advance.

- Tube: X-Ray tube used (Cr, Cu, Fe, Mn ...),
- Wavelength: of the previous tube,
- hkl: Miller indices of the diffraction planes,
- Struct: crystal structure of the sample (BCC, FCC, HCP ...),
- a: crystal lattice parameter,
- c: crystal lattice parameter,
- d_{hkl}: d spacing value,
- Nu: Poisson's ratio,
- E: Young's modulus,
- S1: X-Ray elastic constant,
- S2: X-Ray elastic constant,
- Yield_Stress, and
- Max_R_S: in percentage of the Yield Stress.

These are all of the fields of the material database (columns A through O in Figure [6-2], page 103 . A MetaSlot[9] has been programmed in Nexpert Object which contains an

[9] See footnote page 39.

Order of Source[10] which tells it where to retrieve the data when it needs it (in our case in the MATERIAL.SLK file). This specific MetaSlot constitutes a very good example of the way knowledge can be attached to an object rather than to a specific rule as required in "classical" expert systems.

6.4 External routines

Nexpert Object has been designed to allow easy communication with other processes. It can trigger external actions[11] when it reaches a conclusion. It can also receive events from the external environment, for example the materials database and then modify its internal agenda (inference engine) to process these events. Finally, Nexpert Object is a symbolic processor and is not necessarily the ideal tool for processing numerical data (during statistical analysis for example). Thus, if the application requires complex computations, Nexpert Object can take advantage of optimized external routines or existing math libraries.

We have tested this capability to verify that it works. Indeed the importation of data from the TEC machine is now performed by triggering a C + assembly language program. This routine can be triggered from Nexpert Object if the user has not loaded the experimental data before running the expert system analysis. The data can also be imported from the database management system as described in [Courtney, 1991].

---

[10] See footnote page 39.

[11] Using "Calling-Out", see footnote page 12.

6.5 Graphics presentation

To allow for the best possible interpretation of data it is essential that the user have access to a variety of graphical representations. In the prototype system, it has been verified that Nexpert Object allows utilization of the required graphics programs. Therefore, as a matter of example, several graphic files have been generated and are displayed during the Nexpert Object session. These files are static and have been created using Micrografx Designer and Microsoft Paint[12]. For the present demonstration, during both static and dynamic retrieval of data from Excel a schematic drawing of the data flow through the system is presented to assist the user in understanding the data processing operation.

6.6 User interface

The prototype of the expert system described in this thesis has not been concerned with the final user interface but rather has used the Nexpert Object Development System default interface. This interface is not easy for a non-expert to use and is inappropriate for the final system. However, it is unwise to attempt to develop the user interface at this point because of the remaining uncertainties in the interactive process of the system. The example of the Dipmeter Advisor[13], an expert system

---

[12] Indeed Nexpert Object does not recognize directly the Micrografx Designer format, so the pictures have been designed using Designer with some pictures extracted from the Business collection (one of the Micrografx ClipArt collections available with Designer) and then transferred to Microsoft Paint using the clipboard.

[13] See [Smith, 1983].

developed by Schlumberger to interpret oil well logs measurements, is often quoted to characterize the amounts of code which must be devoted to the different functions:

- Inference Engine: 8 %
- Knowledge Base: 22 %
- Feature Detection: 13 %
- User Interface: 42 %
- Support Environment: 15 %

A more sophisticated and user friendly interface will be developed, if necessary with external C routines or any other specific software, in the future stages of the project. We have evaluated the possibility to customize Excel pull-down menus to realize this user interface. Thus we could trigger the expert system analysis directly from Excel. The same procedure could be used to draw the d-vs-$\sin^2$Psi plots necessary for the final report. The nature of this user interface has been kept in mind during the prototype development in order to make development of the final interface as simple as possible.

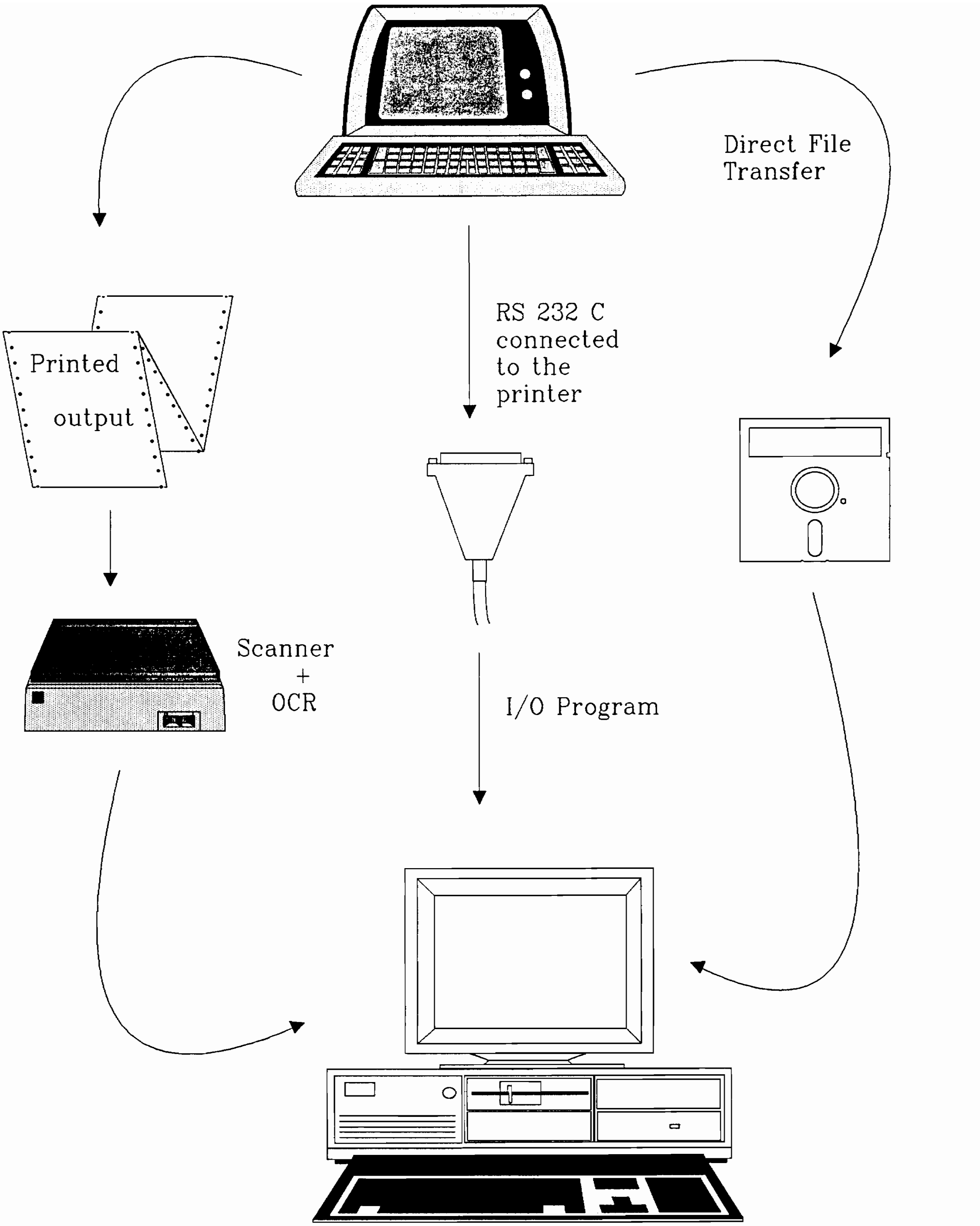
6.7 Callable interface

An important feature of Nexpert Object is its ability to be embedded in an existing software package using the Callable Interface. With it, the "AI part" of the software can be called from the "conventional part" in a transparent way for the user. This solution is used with most of the final versions of Expert Systems using Nexpert Object, and thus will be considered in the next stages of our software.

The current prototype does not use this Callable Interface but according to an engineering consultant[14] a two stage development approach (prototype without using the Callable Interface and final version using it) can be applied and does not demand that the entire knowledge base be rewritten.

---

[14] Vipin Chawla, Ph.D, Senior Knowledge Engineering Consultant, at TAI, has developed a chemical Spill Response Expert System (SRES) for Ciba-Geigy Corporation. This expert system enables all Ciba-Geigy emergency response team members to provide consistent and accurate spill response services in a timely manner by having a knowledge base available on a 24 hours basis. The Phase I prototype was developed for the Microsoft Windows-based user interface using Nexpert Object expert system shell on an IBM PC/AT compatible . Some of the capabilities of this prototype have been enhanced in Phase II such as the integration of the system with a conventional database. Additionally, a device independent user interface for porting this system to a Vax will be developed in the next phase. (Private communication, April 1990).



SYSTEM INTEGRATION - Figure [6-1] Transfer of data

	A	B	C	D	E	F	G	H
1	Sample Description :							
2								
3								
4								
5	Measurement		Type		Date			
6			Chi Value					
7			Phi value				PSI Oscillation Angle Range	
8			Psi Value				Slit Resolution in Two_Theta	
9					Collimator Shape/Size			
10			X-Ray Target Material				Stress Spectra Count Time (Sec)	
11			Wavelength				Peak Bounding Range (%)	
12								
13	Material:							
14		Database		ID #		Name		
15		(1+Mu)/E			Anisotropy Factor			
16								
17	Stress Spectra:							
18			File Specifications			Acquisition Date		
19								
20								
21	Detec	Var	Sin^2(Var)	Intens	FWHM	2-Theta	D-Spacing	St. Dev
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								
33								
34	Results:							
35			D Spacing Intercept				X-Ray Depth Penetration-Microns	
36			Slope of Fitted Line					
37								
38	Stress:		(Ksi)				(MPa)	
39			Value	CS Error		Value	CS Error	
40		Normal						
41		Shear						
42								
43	Error:							
44				Total Statistical Error		Ksi		MPa
45				Estimated Instrumental Error		Ksi		MPa
46								

SYSTEM INTEGRATION - Figure [6-2] Superset with cell names

Material	I.D	Tube	Wavelength	hkl	Struct	a	c	dhkl	Nu	E	S1	S2	Yield St	Max R.S
			Angstrom							MPa	-Nu/E	2(1+Nu)/E	MPa	% Yield
304 Stainless Steel	80	Cr	2.29092	220	BCC				0.30	200000	-1.5E-06	1.30E-05	240	100
410 Stainless Steel Rc 22	83	Cr	2.29092	211	BCC				0.30	200000	-1.5E-06	1.30E-05	275	100
Alumina	199	Cu	1.54178		HEX				0.35	70000	-4.9E-06	3.84E-05	125	100

Figure [6-3] - Materials Database

7. RESULTS AND DISCUSSION

What will happen when all these artificially intelligent computers and robots leave us with nothing to do ? What will be the point of living ? Granted that human obsolescence is hardly an urgent problem. It will be a long, long time before computers can master politics, poetry, or any of the other things we really care about. But a "long time" is not forever; what happens when the computers have mastered politics and poetry ? One can easily envision a future when the world is run quietly and efficiently by a set of exceedingly expert systems, in which machines produce goods, services, and wealth in abundance, and where everyone lives a life of luxury. It sounds idyllic - and utterly pointless.

But personally, I have to side with the optimistic - for two reasons. The first stems from the simple observation that technology is made by people. Despite the strong impression that we are helpless in the face of, say, the spread of automobiles or the more mindless clerical applications of computers, the fact is that technology does not develop according to an immutable genetic code. It embodies human values and human choices... My second reason for being optimistic stems from a simple question: What does it mean to be obsolete ?

M. Mitchell Waldrop.

7.1 Speed problems

During the first part of the development of the prototype we encountered significant execution speed problems using the development version of Nexpert Object under Microsoft Windows. Indeed, even with our 20 Mhz 386 machine with more than 5.5 MBytes of memory (640 KBytes of conventional and 5 MBytes of extended memory), Nexpert Object was very slow. For example, the compilation of a single new rule or the creation of a single new object often required more than 15 or 20 seconds. This problem did not arise because of the high CPU power needed in the AI application, but from the frequent disk accesses. This was largely due to the transformation of Extended memory into Expanded memory using Microsoft Windows/386 and the 640 KBytes barrier. According to an engineering consultant[1] this problem is common using the development version on a PC. For this reason a 2 MByte ramdrive was used to (slightly) improve the performance of the system. This ramdrive contains the following files :

. \windows*.fon

. \windows\win*.*

---

[1] Tom Morgan, Senior Knowledge Engineering Consultant, Technology Applications Incorporated has already developed several Expert Systems using Nexpert Object. He is also responsible for the technical Hot-Line for Nexpert Object for all Bechtel clients. Even with a 386 PC with 11 MBytes of Extended Memory, Windows 286 with a specific utility (Qualita's 386-to-the-Max) to convert extended memory into expanded (this solution has been unofficially recognized as faster than Windows/386 by Microsoft), the system is still too slow for the development stage. For that reason he develops all his knowledge base using a Macintosh and delivers it on a PC with the protected mode runtime system (which give full access to the extended memory and so is as fast as a Macintosh).

. \windows\msdos.exe
. nexpert.exe
. nexpert.dat
. nexpert.ini

i.e. all the files involved in the process. Thus, a bigger ramdrive would not improve the speed of the prototype.

Most of these problems have now been solved using version 3.0 of Microsoft Windows. We have already checked that, with a platform such as the Macintosh which does not have any memory barrier and thus is free of these memory problems, our prototype has a normal response time even with the development system version of Nexpert Object. Therefore we believe that these problems should not affect the runtime version, i.e the end user version of our system.

The utilization of a 486 machine during the next development stage of the project is also being considered.

7.2 Prototype evaluation

Contrary to all the conventional programs, for which a variety of debugging techniques are available, verification of Artificial Intelligence "programs" is still a research subject[2].

Therefore we have chosen to check the validity of our knowledge base using a wide range of measurements made on different materials and related to various well known problems such as large grain size or preferred orientation.

---

2 For a complete description of the different techniques available see [Hite, 1988].

A few examples of deliberately invalid data were created (modifying some real data with Excel) to trigger validation data related rules in the knowledge base. Our expert system was able to diagnose several important problems in the experimental data.

8. CONCLUSIONS

We know what we are, but know not what we may be.
William Shakespeare.

I never think of the future, it comes soon enough.
Albert Einstein.

What is possible we can do now, what is impossible will take a little longer.
A modern-day proverb.

The problems of the world cannot possibly be solved by skeptics or cynics whose horizons are limited by the obvious realities. We need men who can dream of things that never were.
John F. Kennedy

When all else fails, the future still remains.
Christian Bovee.

It is hard to predict - especially the future.
Niels Bohr, Physicist.

The present work shows that all the concepts needed for a full scale version of an expert system for the validation and interpretation of X-ray residual stress data have been successfully tested. Therefore we are now convinced of the feasibility of such a project, and the foundations on which to build it has been designed.

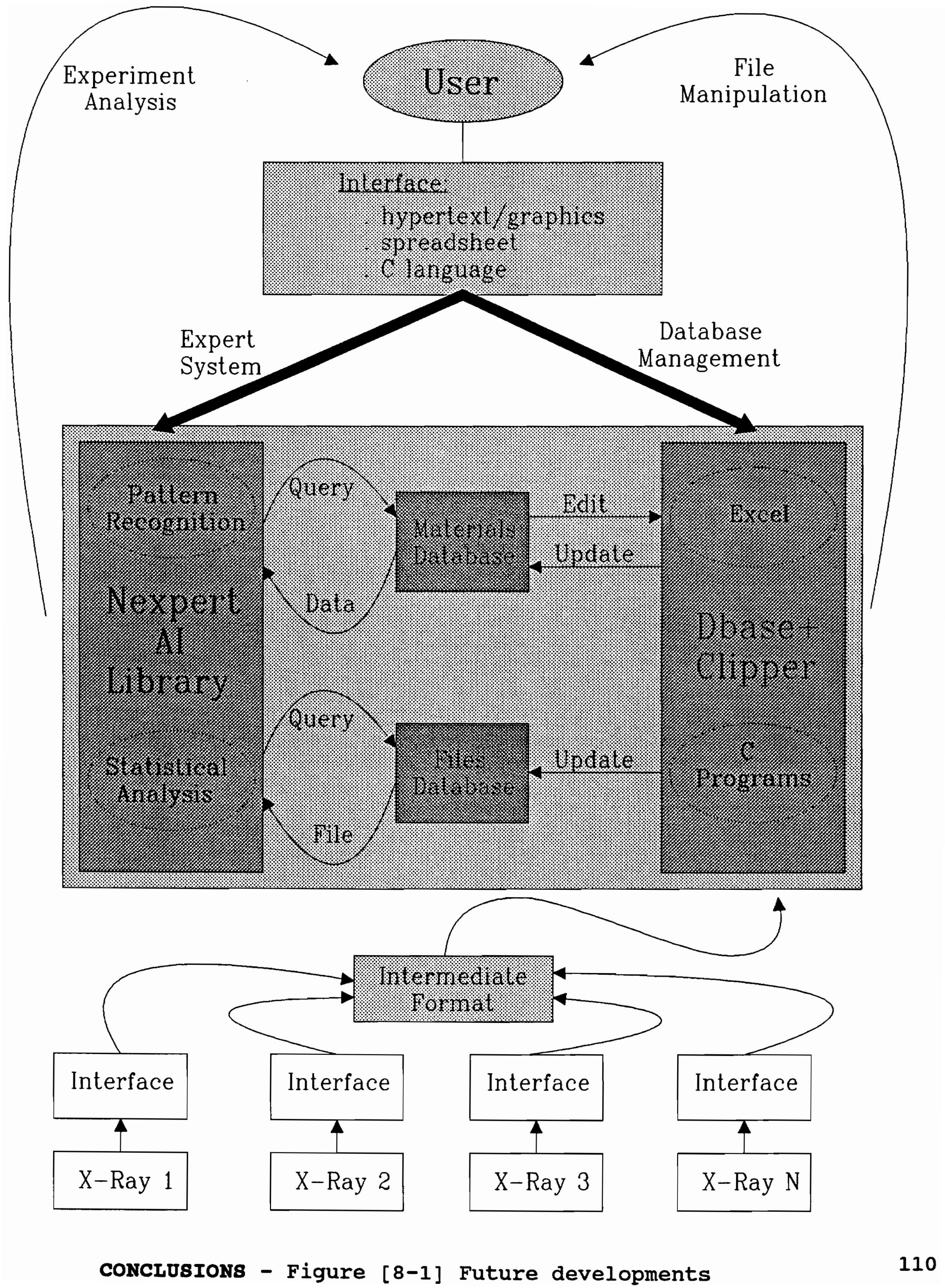
Thus, the future work which must be done is clearly :

- continue the knowledge acquisition to interpret and classify the measurements. This part will require extensive use of statistical analysis as a way to discriminate data. Indeed the major part of the validation of the data has now been implemented and we should focus on the discrimination between the various categories previously shown,

- develop a friendly user interface to allow a non-computer-familiar technician to have access to the full power of the expert system. We are convinced that such a user interface can be built using the custom pull-down menu capabilities now available in Excel 3.0, and

- test and check the consistency of the expert system in the field against industrial problems, thus allowing us to "fine-tune" the rules in our expert system.

An overview of what our system should look like when completed is shown on Figure [8-1] on the next page.



CONCLUSIONS - Figure [8-1] Future developments

9. APPENDICES

What if these theories are really true, and we were magically shrunk and put into someone's brain while he was thinking. We would see all the pumps, pistons, gears and levers working away, and we would be able to describe their working completely, in mechanical terms, thereby completely describing the thought process of the brain. But that description would nowhere contain any mention of thought ! It would contain nothing but descriptions of pumps, pistons, levers !
Wilhelm Leibniz, commenting on theories that the brain was "just" a complicated mechanical computer.

9.1 List of rules

The next pages contain a comprehensive list of the rules developed for our knowledge base. It should be noted that these examples do not follow the exact Nexpert Object syntax, but are an "english-like" presentation.

```
RULE :  Rule All_Variable_Angles_Negative (#1)
If
          there is evidence of Measurement_Type_Updated
          And MAX(<|Angles|>.Psi) is less than or equal
          to 0
Then All_Variable_Angles_Negative
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=All your
          @V(Variable_Angle) angles are negative, we
          recommand to measure with both negative and
          positive @V(Variable_Angle)  angles.,@ADD";)
```

```
RULE :  Rule All_Variable_Angles_Positive (#2)
If
          there is evidence of Measurement_Type_Updated
          And MIN(<|Angles|>.Psi) is greater than or
equal to 0
Then All_Variable_Angles_Positive
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=All your
          @V(Variable_Angle) angles are positive, we
          recommand to measure with both negative and
          positive @V(Variable_Angle) angles.,@ADD";)
```

```
RULE :  Rule Average_Standart_Deviation_Too_Big (#3)
If
          Average.St_Dev is greater than 0.0001
Then Average_Standart_Deviation_Too_Big
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The average of the
          standart deviation of your measurements is too
          big (= @V(Average.St_Dev)), we recommand that
          your increase your stress spectra count time
          (current value ="
```

```
RULE :  Rule Configuration_Changed (#4)
If
          there is no evidence of Answer.Configuration
          And Reset Answer.Configuration
          And X_Ray_Machine.Model is "PARS 1610-2","SET-
          X","Huber-4030","XRD-3000 Type
          PTS","Canmet/Proto S.","Mexstress","D1000"
          And X_Ray_Machine.Goniometry is
          "Chi/Omega","Psi"
          And STRUPPER(Subdirectory.Program) is assigned
          to Subdirectory.Program
          And STRUPPER(Subdirectory.Measurement) is
          assigned to Subdirectory.Measurement
Then Configuration.Changed
          is confirmed.
          And Writing in "XPSETUP.NXP"
          @TYPE=NXP;@FILL=NEW;@ATOMS=X_Ray_Machine.Model,
          X_Ray_Machine.Goniometry,Subdirectory.Program,S
          ubdirectory.Measurement;
          And Reset Configuration.Incorrect
```

```
RULE :  Rule Configuration_Incorrect (#5)
If
          Show "XPSETUP.NXP"
          @KEEP=FALSE;@WAIT=FALSE;@RECT=95,325,500,150;
          And Retrieve "XPSETUP.NXP"
          @TYPE=NXP;@NAME="|Controls|";
          And Answer.Configuration is assigned to
          Answer.Configuration
          And there is no evidence of
          Answer.Configuration
          And Reset Configuration.Changed
Then Configuration.Incorrect
          is confirmed.
          And Reset X_Ray_Machine.Model
          And Reset X_Ray_Machine.Goniometry
          And Reset Subdirectory.Program
          And Reset Subdirectory.Measurement
```

```
RULE :  Rule D_Spacing_Ok (#6)
If
          Angle_Number.D_Spacing_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
```

```
          And ABS(Average.D_Spacing-
          'Angle_'\Angle_Number.D_Spacing_Counter\.D_Spac
          ing)-Range.D_Spacing is less than 0
Then D_Spacing_Ok
          is confirmed.
          And Reset D_Spacing_Ok
          And Reset D_Spacing_Too_Big
          And Reset D_Spacing_Too_Small
          And
          'Angle_'\Angle_Number.D_Spacing_Counter\.Psi is
          assigned to Temp.Floa
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The D_Spacing #
          @V(Angle_Number.D_Spacing_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is
          OK.,@ADD";)
          And Angle_Number.D_Spacing_Counter+1 is
          assigned to Angle_Number.D_Spacing_Counter
```

```
RULE :  Rule D_Spacing_Too_Big (#7)
If
          Angle_Number.D_Spacing_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
          And
          'Angle_'\Angle_Number.D_Spacing_Counter\.D_Spac
          ing-Average.D_Spacing-Range.D_Spacing is
          greater than 0
Then D_Spacing_Too_Big
          is confirmed.
          And Reset D_Spacing_Ok
          And Reset D_Spacing_Too_Big
          And Reset D_Spacing_Too_Small
          And
          'Angle_'\Angle_Number.D_Spacing_Counter\.Psi is
          assigned to Temp.Floa
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The D_Spacing #
          @V(Angle_Number.D_Spacing_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is too
          big.,@ADD";)
          And Angle_Number.D_Spacing_Counter+1 is
          assigned to Angle_Number.D_Spacing_Counter
```

```
RULE :  Rule D_Spacing_Too_Small (#8)
If
          Angle_Number.D_Spacing_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
          And Average.D_Spacing-
          'Angle_'\Angle_Number.D_Spacing_Counter\.D_Spac
          ing-Range.D_Spacing is greater than 0
Then D_Spacing_Too_Small
          is confirmed.
          And Reset D_Spacing_Ok
          And Reset D_Spacing_Too_Big
          And Reset D_Spacing_Too_Small
          And Reset D_Spacing_Ok
          And
          'Angle_'\Angle_Number.D_Spacing_Counter\.Psi is
          assigned to Temp.Floa
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The D_Spacing #
          @V(Angle_Number.D_Spacing_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is too
          small.,@ADD";)
          And Angle_Number.D_Spacing_Counter+1 is
          assigned to Angle_Number.D_Spacing_Counter
```

```
RULE :  Rule Experiment_File_Converted (#9)
If
          there is no evidence of
          Experiment_File.PC_To_Sylk
          And STRCAT(Subdirectory.Program,"ASC2SLK.HLP")
          is assigned to Temp.Strin
          And Show "@V(Temp.Strin)"
          @KEEP=FALSE;@WAIT=TRUE;@RECT=95,325,500,150;
Then Experiment_File.Converted
          is confirmed.
          And STRCAT(Subdirectory.Program,"ASC2SLK.EXE")
          is assigned to Temp.Strin
          And Execute
          "@V(Temp.Strin)"(@TYPE=EXE;@WAIT=TRUE;)
          And Experiment_File.PC_To_Sylk is set to TRUE
```

```
RULE :  Rule Experiment_File_Do_Not_Exist (#10)
If
          there is evidence of
          Experiment_File.Transferred_And_Converted
          And STRUPPER(File_Name) is assigned to
          File_Name
          And
          SUBSTRING(File_Name,0,CHARFIND(File_Name,"."))
          is assigned to File_Name
          And STRCAT(Subdirectory.Measurement,File_Name)
          is assigned to File_Name
          And STRCAT(File_Name,".TXT") is assigned to
          Experiment_File.Analysis
          And STRCAT(File_Name,".SLK") is assigned to
          File_Name
          And Execute
          "FileExist"(@STRING="@FILE=@V(File_Name),@RETUR
          N=Experiment_File.Exist";)
          And there is no evidence of
          Experiment_File.Exist
Then Experiment_File.Do_Not_Exist
          is confirmed.
          And Execute "Message"(@STRING="@TEXT=The file
          @V(File_Name) do not exist.,@OK";)
          And Reset File_Name
          And Reset Experiment_File.Exist
          And Reset Experiment_File.Do_Not_Exist
          And Experiment_File.Transferred_And_Converted
          is set to TRUE
```

```
RULE :  Rule Experiment_File_Transferred (#11)
If
          there is no evidence of
          Experiment_File.X_Ray_To_PC
          And STRCAT(Subdirectory.Program,"TEC2ASC.HLP")
          is assigned to Temp.Strin
          And Show "@V(Temp.Strin)"
          @KEEP=FALSE;@WAIT=TRUE;@RECT=95,325,500,150;
Then Experiment_File.Transferred
          is confirmed.
          And STRCAT(Subdirectory.Program,"TEC2ASC.EXE")
          is assigned to Temp.Strin
          And Execute
          "@V(Temp.Strin)"(@TYPE=EXE;@WAIT=TRUE;)
          And Experiment_File.PC_To_Sylk is set to FALSE
          And Experiment_File.X_Ray_To_PC is set to TRUE
          And Reset Experiment_File.Converted
```

```
RULE :  Rule Experiment_File_Transferred_And_Converted
(#12)
If
          there is no evidence of Configuration.Incorrect
          And STRCAT(Subdirectory.Program,"LOAD.MSP") is
          assigned to Temp.Strin
          And Show "@V(Temp.Strin)"
          @KEEP=FALSE;@WAIT=FALSE;@RECT=5,325,250,150;
          And Experiment_File.X_Ray_To_PC is assigned to
          Experiment_File.X_Ray_To_PC
          And there is evidence of
          Experiment_File.X_Ray_To_PC
          And Experiment_File.PC_To_Sylk is assigned to
          Experiment_File.PC_To_Sylk
          And there is evidence of
          Experiment_File.PC_To_Sylk
Then Experiment_File.Transferred_And_Converted
          is confirmed.
```

```
RULE :  Rule Maximun_Variable_Angle_Too_Small (#13)
If
          MAX(<|Angles|>.Psi) is assigned to Temp.Floa
          And Temp.Floa is less than or equal to 15
Then Maximun_Variable_Angle_Too_Small
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=Your biggest
          @V(Variable_Angle) value is only @V(Temp.Floa)
          degrees. You should try to increase this
          number, we recommand to measure at least from -
          30 to +30 degrees.,@ADD";
```

```
RULE :  Rule Measurement_Loaded (#14)
If
          there is evidence of Experiment_File.Exist
          And STRCAT(Subdirectory.Program,"RETRIEVE.MSP")
          is assigned to Load_Materials_Database_Picture
          And STRCAT(Subdirectory.Program,"LOAD.MSP") is
          assigned to Temp.Strin
          And Show "@V(Temp.Strin)"
          @KEEP=FALSE;@WAIT=FALSE;@RECT=95,325,500,150;
          And Retrieve "@V(File_Name)"
```

```
        @TYPE=SYLK;@NAME="|Reports|";
Then Measurement_Loaded
        is confirmed.
```

```
RULE :  Rule Measurement_Type_Chi_Omega (#15)
If
        there is evidence of Measurement_Type_Updated
        And Measurement.Type is precisely equal to
        "Chi/Omega"
Then Measurement_Type_Chi_Omega
        is confirmed.
        And Reset Measurement.Phi_Value
        And Reset Measurement.Psi_Value
        And Measurement.Chi_Value is set to NOTKNOWN
        And Measurement.Phi_Value is assigned to
        Measurement.Phi_Value
        And Measurement.Psi_Value is assigned to
        Measurement.Psi_Value
```

```
RULE :  Rule Measurement_Type_Psi (#16)
If
        there is evidence of Measurement_Type_Updated
        And Measurement.Type is precisely equal to
        "Psi"
Then Measurement_Type_Psi
        is confirmed.
        And Reset Measurement.Chi_Value
        And Reset Measurement.Phi_Value
        And Measurement.Chi_Value is assigned to
        Measurement.Chi_Value
        And Measurement.Phi_Value is assigned to
        Measurement.Phi_Value
        And Measurement.Psi_Value is set to NOTKNOWN
```

```
RULE :  Rule Measurement_Type_Updated (#17)
If
        there is evidence of Measurement_Loaded
Then Measurement_Type_Updated
        is confirmed.
        And X_Ray_Machine.Goniometry is assigned to
        Measurement.Type
        And X_Ray_Machine.Model is assigned to
```

```
          Material.Database
          And SUBSTRING(Measurement.Type,0,3) is assigned
          to Variable_Angle
```

```
RULE :  Rule Minimun_Variable_Angle_Too_Small (#18)
If
          MIN(<|Angles|>.Psi) is assigned to Temp.Floa
          And Temp.Floa is greater than or equal to -15
Then Minimun_Variable_Angle_Too_Small
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=Your smallest
          @V(Variable_Angle) value is @V(Temp.Floa)
          degrees. You should try to decrease this
          number, we recommand to measure at least from -
          30 to +30 degrees.,@ADD";)
```

```
RULE :  Rule Multiple_Peaks (#19)
If
          there is evidence of
          Statistical_Evaluation_Ready
          And
          Average.D_Spacing/Standart_Deviation.D_Spacing
          is assigned to Temp.Floa
          And Temp.Floa is less than 500
Then Multiple_Peaks
          is confirmed.
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The ratio
          (Average/Standart deviation) or your d_spacing
          is @V(Temp.Floa) suggesting that you are facing
          a multiple peaks problem.,@ADD";)
```

```
RULE :  Rule Number_Of_Variable_Angle_Too_Small (#20)
If
          Number_Of_Variable_Angle is less than or equal
          to 6
Then Number_Of_Variable_Angles_Too_Small
          is confirmed.
          And Execute
```

```
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=You are using only
          @V(Number_Of_Variable_Angle) differents
          @V(Variable_Angle) angles. You should try to
          increase this number.,@ADD";)
```

```
RULE :  Rule Standart_Deviation_Ok (#21)
If
          Angle_Number.St_Dev_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
          And ABS(Average.St_Dev-
          'Angle_'\Angle_Number.St_Dev_Counter\.St_Dev)-
          Range.St_Dev is less than 0
Then Standart_Deviation_Ok
          is confirmed.
          And Reset Standart_Deviation_Ok
          And Reset Standart_Deviation_Too_Big
          And Reset Standart_Deviation_Too_Small
          And 'Angle_'\Angle_Number.St_Dev_Counter\.Psi
          is assigned to Temp.Floa
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The standart deviation
          # @V(Angle_Number.St_Dev_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is
          OK.,@ADD";)
          And Angle_Number.St_Dev_Counter+1 is assigned
          to Angle_Number.St_Dev_Counter
```

```
RULE :  Rule Standart_Deviation_Too_Big (#22)
If
          Angle_Number.St_Dev_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
          And
          'Angle_'\Angle_Number.St_Dev_Counter\.St_Dev-
          Average.St_Dev-Range.St_Dev is greater than 0
Then Standart_Deviation_Too_Big
          is confirmed.
          And Reset Standart_Deviation_Ok
          And Reset Standart_Deviation_Too_Big
          And Reset Standart_Deviation_Too_Small
          And 'Angle_'\Angle_Number.St_Dev_Counter\.Psi
          is assigned to Temp.Floa
```

```
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The standart deviation
          # @V(Angle_Number.St_Dev_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is too
          big.,@ADD";)
          And Angle_Number.St_Dev_Counter+1 is assigned
          to Angle_Number.St_Dev_Counter
```

```
RULE :  Rule Standart_Deviation_Too_Small (#23)
If
          Angle_Number.St_Dev_Counter-
          Number_Of_Variable_Angle is less than or equal
          to 0
          And Average.St_Dev-
          'Angle_'\Angle_Number.St_Dev_Counter\.St_Dev-
          Range.St_Dev is greater than 0
Then Standart_Deviation_Too_Small
          is confirmed.
          And Reset Standart_Deviation_Ok
          And Reset Standart_Deviation_Too_Big
          And Reset Standart_Deviation_Too_Small
          And 'Angle_'\Angle_Number.St_Dev_Counter\.Psi
          is assigned to Temp.Floa
          And Execute
          "WriteTo"(@STRING="@TRANSCRIPT,@FILE=@V(Experim
          ent_File.Analysis),@TEXT=The standart deviation
          # @V(Angle_Number.St_Dev_Counter)
          (@V(Variable_Angle)=@V(Temp.Floa)) is
          OK.,@ADD";)
          And Angle_Number.St_Dev_Counter+1 is assigned
          to Angle_Number.St_Dev_Counter
```

```
RULE :  Rule Statistical_Evaluation_Ready (#24)
If
          there is evidence of Measurement_Loaded
          And <|Angles|>.Intens is not equal to 0
          And LENGTH(<|Angles|>) is assigned to
          Number_Of_Variable_Angle
          And 1 is assigned to
          Angle_Number.D_Spacing_Counter
          And 1 is assigned to
          Angle_Number.St_Dev_Counter
Then Statistical_Evaluation_Ready
          is confirmed.
```

```
And Create Object 'Material_'\Material.Id\
|Materials|
And AVERAGE(<|Angles|>.Intens) is assigned to
Average.Intens
And AVERAGE(<|Angles|>.FWHM) is assigned to
Average.FWHM
And AVERAGE(<|Angles|>.D_Spacing) is assigned
to Average.D_Spacing
And AVERAGE(<|Angles|>.St_Dev) is assigned to
Average.St_Dev
And STDEV(<|Angles|>.Intens) is assigned to
Standart_Deviation.Intens
And STDEV(<|Angles|>.FWHM) is assigned to
Standart_Deviation.FWHM
And STDEV(<|Angles|>.D_Spacing) is assigned to
Standart_Deviation.D_Spacing
And 2*Standart_Deviation.D_Spacing is assigned
to Range.D_Spacing
And STDEV(<|Angles|>.St_Dev) is assigned to
Standart_Deviation.St_Dev
And 2*Standart_Deviation.St_Dev is assigned to
Range.St_Dev
```

9.2 List of classes, objects and properties

9.2.1 Character based presentation

9.2.1.1 List of classes

Following is a comprehensive list in alphabetical order of all the classes created in our prototype :

Angles
Controls
Descriptions
Materials
Results

9.2.1.2 List of objects

Following is a comprehensive list in alphabetical order of all the objects created in our prototype :

All_Variable_Angles_Negative
All_Variable_Angles_Positive
Angle_1
Angle_10
Angle_2
Angle_3
Angle_4
Angle_5
Angle_6
Angle_7
Angle_8
Angle_9

Angle_Number
Answer
Average
Average_Standart_Deviation_Too_Big
Configuration
D_Spacing_Ok
D_Spacing_Too_Big
D_Spacing_Too_Small
Error
Experiment_File
File_Name
Load_Materials_Database_Picture
Material
Maximun_Variable_Angle_Too_Small
Measurement
Measurement_Loaded
Measurement_Type_Chi_Omega
Measurement_Type_Psi
Measurement_Type_Updated
Minimun_Variable_Angle_Too_Small
Multiple_Peaks
Number_Of_Variable_Angle
Number_Of_Variable_Angles_Too_Small
Range
Result
Sample_Description
Standart_Deviation
Standart_Deviation_Ok
Standart_Deviation_Too_Big
Standart_Deviation_Too_Small
Statistical_Evaluation_Ready
Stress
Stress_Spectra
Subdirectory

Temp
Variable_Angle
X_Ray_Machine

9.2.1.3 List of properties

Following is a comprehensive list in alphabetical order of all the properties created in our prototype :

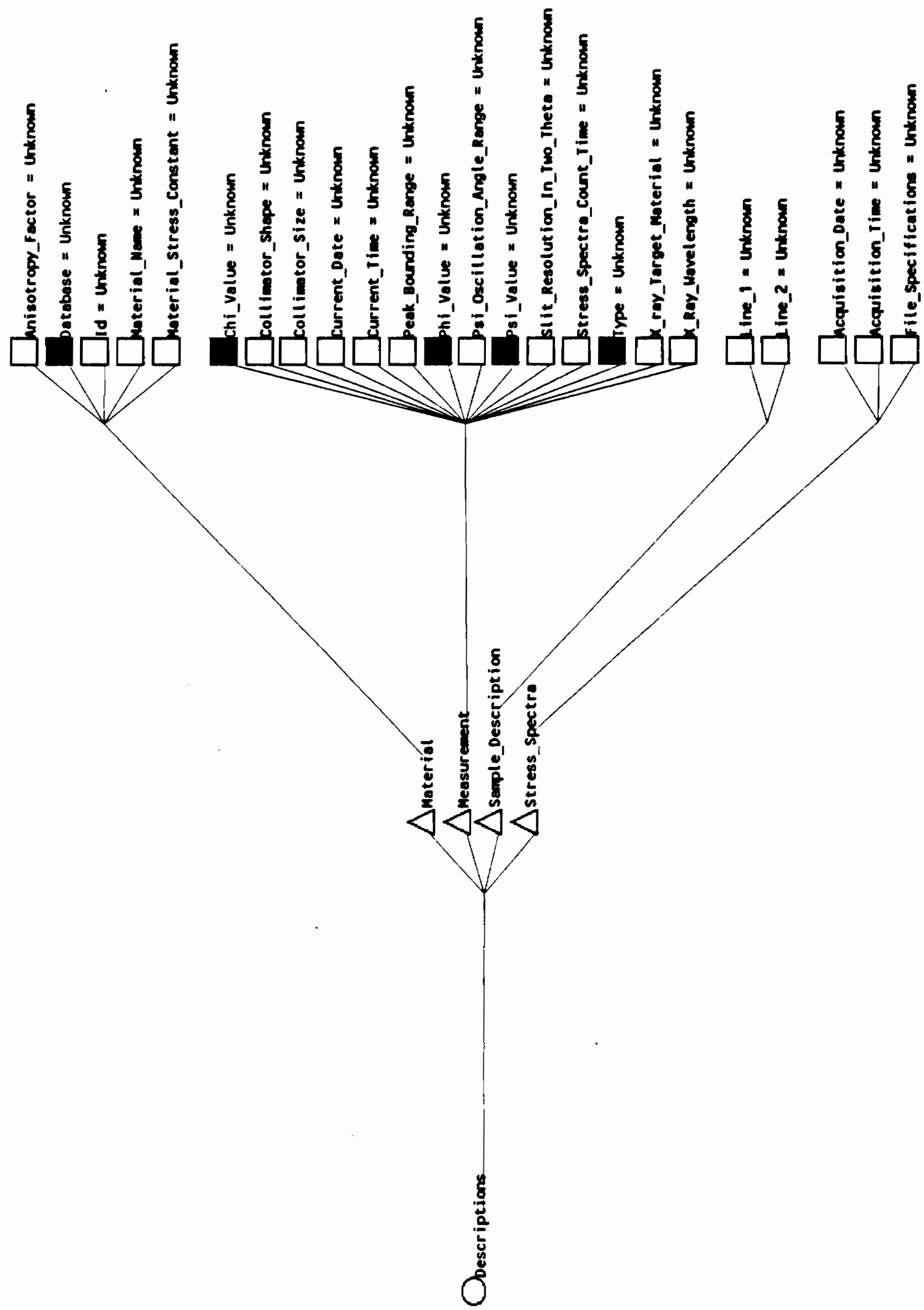
a
Acquisition_Date
Acquisition_Time
Analysis
Anisotropy_Factor
c
Changed
Chi_Value
Collimator_Shape
Collimator_Size
Configuration
Converted
Current_Date
Current_Time
D_Spacing
D_Spacing_Counter
D_Spacing_Intercept
Database
Designation
Detector
dhkl
Do_Not_Exist
E
Estimated_Instrumental_Error_Ksi

Estimated_Instrumental_Error_MPa
Exist
File_Specifications
Floa
FWHM
Goniometry
hkl
Id
Incorrect
Intege
Intens
Line_1
Line_2
Material_Name
Material_Stress_Constant
Maximun_Residual_Stress
Measurement
Model
Normal_CS_Error_Ksi
Normal_CS_Error_MPa
Normal_Ksi
Normal_MPa
Nu
PC_To_Sylk
Peak_Bounding_Range
Phi_Value
Program
Psi
Psi_Oscillation_Angle_Range
Psi_Value
S1
S2
Shear_CS_Error_Ksi
Shear_CS_Error_MPa

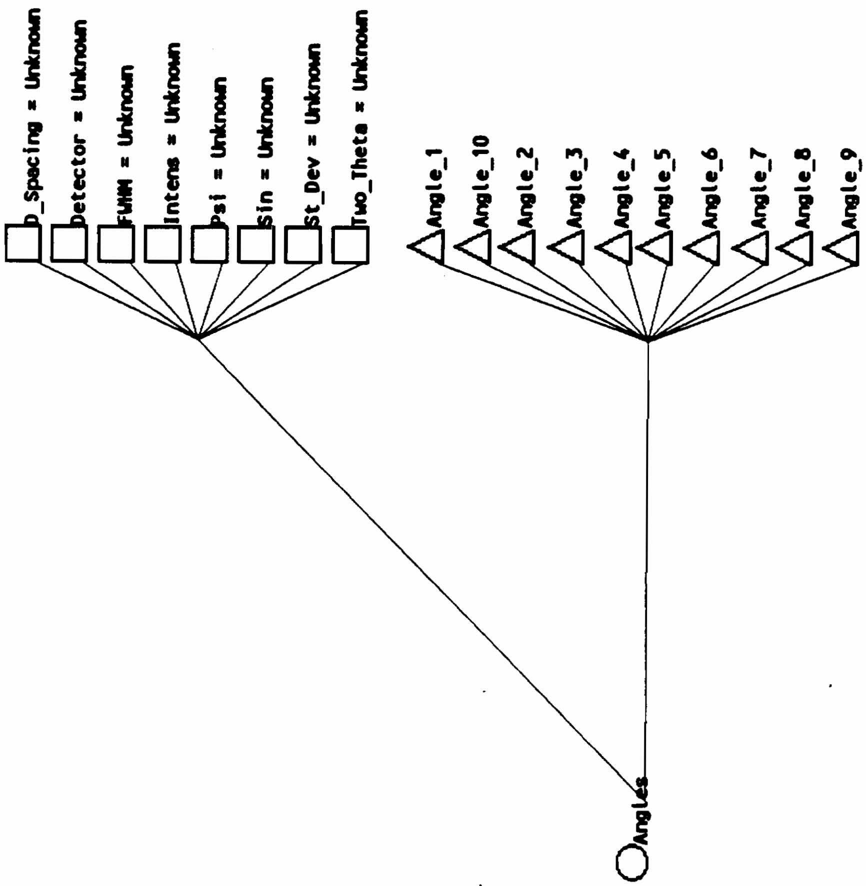
Shear_Ksi
Shear_MPa
Sin
Slit_Resolution_In_Two_Theta
Slope_Of_Fitted_Line
St_Dev
St_Dev_Counter
Stress_Spectra_Count_Time
Strin
Structure
Total_Statistical_Error_Ksi
Total_Statistical_Error_MPa
Transferred
Transferred_And_Converted
Tube
Two_Theta
Type
Value
X_Ray_Depth_Penetration
X_ray_Target_Material
X_Ray_To_PC
X_Ray_Wavelength
Yield_St

9.2.2 Graphical presentation

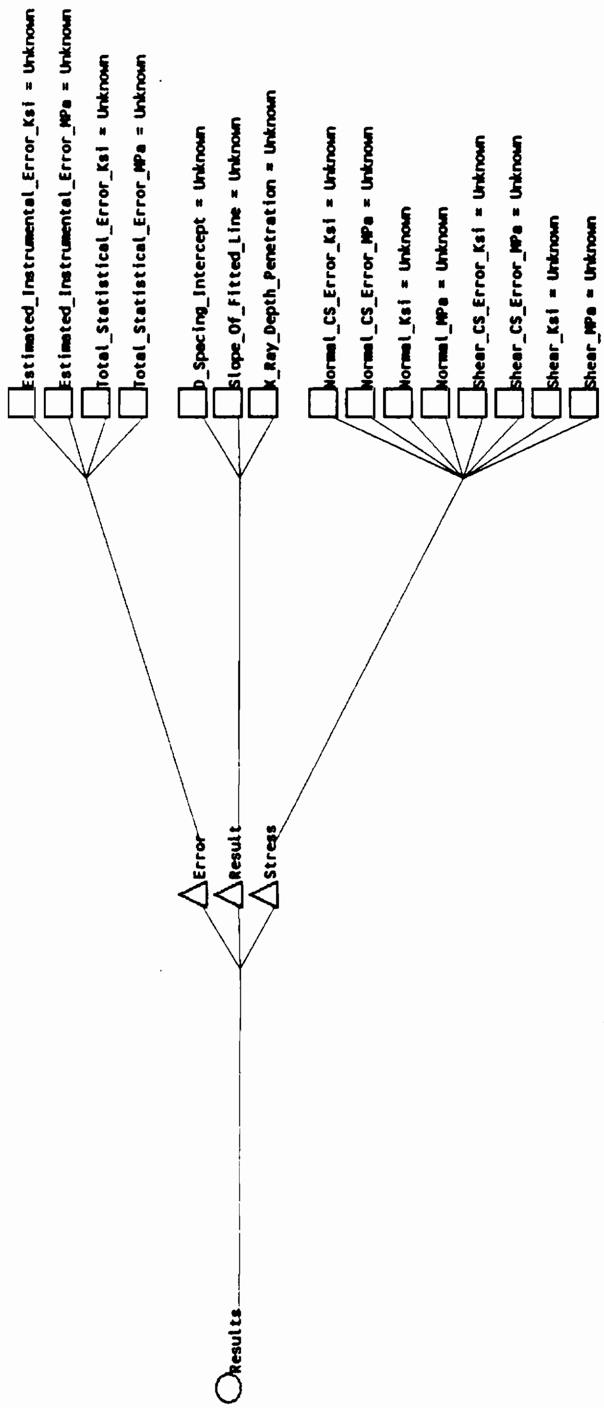
The following charts are the direct output of the graphical representation of the object network prepared by Nexpert Object. Unfortunately, this program does not provide any user control over the format of the output. It is recognized that the following pages will be difficult to read but they are the only graphical documentation available.



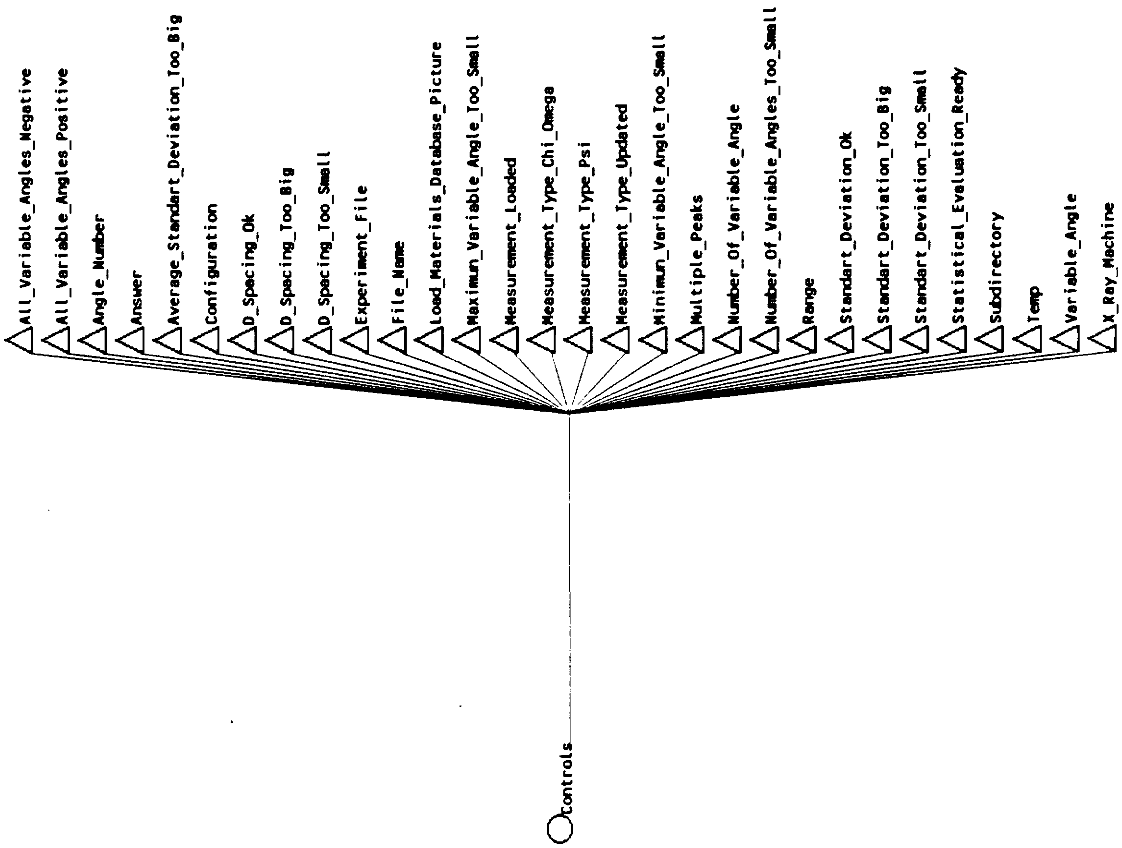
APPENDICES - Figure [9-1] Class Descriptions



APPENDICES - Figure [9-2] Class Angles



APPENDICES - Figure [9-3] Class Results



APPENDICES - Figure [9-4] Class Controls

10. BIBLIOGRAPHY

[Aiken, 1990] Milam W. Aiken, Olivia R. Liu Sheng, "Nexpert Object Software Review", Expert Systems, February 1990, Vol. 7, No. 1, pp. 54-57.

[Aleksander, 1987] Igor Aleksander and Piers Burnett, "What's the Problem ?", in Thinking Machines: The search for Artificial Intelligence, 1987, pp. 58-67.

[Balasubramaniam, 1990] A. Balasubramaniam, Guturu Parthasarathy and B. N. Chatterji, "Knowledge based approach to cluster algorithm selection", in Pattern Recognition Letters, 1990, Vol. 11, No. 10, pp. 651-661.

[Bechtel, 1990] Bechtel AI Institute, "Nexpert Object - Technical Notes", 1990.

[Boden, 1990] Margaret A. Boden, "Escaping from the Chinese room", in The Philosophy of Artificial Intelligence, 1990, pp. 89-104.

[Boley, 1990] Harold Boley, "Expert System shells: very high languages for Artificial Intelligence", in Expert Systems, February 1990, Vol. 7, No. 1, pp. 2-8.

[Brand, 1987] Stewart Brand, The Media Lab - Inventing the Future at M.I.T, Penguin Books, 1987.

[Brown, 1989] D. J. H. Brown, "A Knowledge Acquisition Tool for Decision Support Systems", in SIGART Newsletter, April 1989, No. 108, pp. 93-97.

[Buchanan, 1982] B. Buchanan and R. Duda, "Principles of Rule Based Expert Systems", Stanford University, Dept of Computer Science, HPP-82-14, Aug 1982, 55 pages.

[Carpenter, 1989] Gail A. Carpenter, "Neural Network Models for Pattern Recognition and Associative Memory", in Neural Networks, Vol. 2, pp. 243-257, 1989.

[Carrico, 1989] Michael A. Carrico, John E. Girard and Jennifer P. Jones, Building Knowledge Systems - Developing & Managing Rule - Based Applications, McGraw-Hill Book Company, 1989.

[Castex, 1981] L. Castex, J.L. Lebrun, G. Maeder and J.M Sprauel, Détermination des Contraintes Résiduelles par diffraction des Rayons X, Publications Scientifiques et Techniques, ENSAM Paris, 151, Bd de l'Hopital, 75640, Paris Cedex 13, France.

[Chandra, 1990] Mahesh Chandra and Shivaji Rao, "Artificial Intelligence: The Emerging Corporate Knowledge Frontier", in Global Corporate Intelligence - Opportunities, Technologies and threats in the 1990s.

[Cowan, 1988] Jack D. Cowan and David H. Sharp, "Neural Nets and Artificial Intelligence", in Daedalus - Journal of the American Academy of Arts and Sciences, Winter 1988, pp. 85-121.

[Courtney, 1990] Scott Courtney, Marc Tricard and Robert W. Hendricks, "A Database Management System for X-Ray Residual Stress Data", Progress Report, Oct. 1990.

[Courtney, 1991] Scott Courtney, A Database Management System for X-ray Residual Stress Data, Senior Project Report, Virginia Polytechnic Institute and State University, April 1991.

[Cybenko, 1990] G. Cybenko, "Complexity Theory of Neural Networks and Classification Problems", in Lecture Notes in Computer Science, Vol. 412, pp. 26-44.

[Dehan, 1989-1] Dehan, C.F., Hendricks, R.W., and Roach, J.W., "An Intelligent Workstation for Stress Determination Using X-Ray Diffraction", in International Conference on Residual Stresses --- ICRS2, G. Beck, S. Denis, and A. Simon, eds., Nancy, France, 22-26 November, 1988, London: Elsevier Applied Science, pp. 39-45, (1989).

[Dehan, 1989-2] Dehan, C.F., An Intelligent Workstation for Reliable Residual Stress Determination Using X-Ray Diffraction, M.S. Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, (July 1989).

[Dennet, 1990] Daniel C. Dennet, "Cognitive Wheels: the Frame Problem of AI", in The Philosophy of Artificial Intelligence, 1990, pp. 147-170.

[Drake, 1990] Keith C. Drake and Paul Hess, "Abduction: A Numeric Knowledge Acquisition Approach", in PC AI, Sep/Oct 1990, pp. 58-61.

[Drape, 1990] Gaylen Drape, Larry Lafferty and Suzanne Leonard, "Developing PC-Based Expert Systems -- Promises Fulfilled and Lessons Learned", in SPIE Vol. 1293 Applications of Artificial Intelligence VIII, 1990, pp. 484-493.

[Durkin, 1990] John Durkin, "Introducing students to expert systems", in Expert Systems, May 1990, Vol. 7, No. 2, pp. 70-80.

[Dym, 1991] Clive L. Dym and Raymond E. Levitt, Knowledge-Based Systems in Engineering, Mc Graw-Hill, Inc, 1991.

[Feigenbaum, 1988] Edward Feigenbaum, Pamela McCorduck and H. Penny Nii, The Rise of the Expert System Company, Times Book., 1988.

[Fenves, 1986] S. J. Fenves, "What is an Expert System?", Expert Systems in Civil Engineering, edited by C. N. Kostem and M. L. Maher, ASCE, 1986, pp. 1-6.

[Fikes, 1985] R. Fikes and T. Kehler, "The Role of Frame Based Representation in Reasoning", Communications of the ACM, Vol. 28, No. 9, Sep 1985, pp. 904-920.

[Frank, 1988-1] S. J. Frank, "What AI Practitioners Should Know about the Law, Part 1", The AI Magazine, Vol. 9, No. 1, Spring 1988, pp. 63-75.

[Frank, 1988-2] S. J. Frank, "What AI Practitioners Should Know about the Law, Part 2", The AI Magazine, Vol. 9, No. 2, Summer 1988, pp. 109-114.

[Freiling, 1985] M. Freiling at al., "Starting a Knowledge Engineering Project: A Step by Step Approach", The AI Magazine, Vol. 6, No. 3, Fall 1985, pp. 150-164.

[Fox, 1990] Mark S. Fox, "AI and Expert System : Myths, Legends and Facts", IEEE Expert, February 1990, pp. 8-20.

[Genesereth, 1985] M. R. Genesereth and M. L. Ginsberg, "Logic Programming", Communications of the ACM, Vol. 28, No 9, Sep 1985, pp. 933-941.

[Gilmore, 1985] J.F. Gilmore and K. Pulaski, "A survey of Expert Systems Tools", IEEE 1985 Second Conference on Artificial Intelligence Applications, pp. 498-502.

[Gilmore, 1986] J.F Gilmore, C. Howard and K. Pulaski, "A comprehensive evaluation of the expert system tools", in the proceedings of Applications of Artificial Intelligence IV, pp. 194-208, 15-16 April 1986, Innsbruck, Austria.

[Gorman, 1988] R. Paul Gorman and Terrence J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets", in Neural Networks, Vol. 1, pp. 75-79, 1988.

[Halacy, 1962] D. S. Halacy, Jr, Computers - The machines we think with, Dell Publishing Co., Inc, 1962.

[Hart, 1985] A. Hart, "Knowledge Elicitation: Issues and Methods", CAD, Vol. 17, No. 9,. Nov 1985, pp. 455-462.

[Hayes-Roth, 1987] F. Hayes-Roth, "Expert Systems" in Encyclopedia of Artificial Intelligence, Vol 1, pp. 287-298.

[Hekmatpour, 1990] Amir Hekmatpour and Paul Chau, "AI Techniques and Object Oriented Technology for VLSI Design-Space Representation, Optimization and Management, in SPIE Vol. 1293 Applications of Artificial Intelligence VIII,1990, pp.85-94.

[Hendricks, 1991] Robert W. Hendricks, "An Hypertext System for the Operation of X-Ray Residual Stress Apparatus", Unpublished Research Proposal, Feb. 1991.

[Hite, 1988] Lee Anne Hite, Designing a Testing Strategy for Expert Systems, M.S thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, (February 1988).

[Hoffman, 1987] Robert R. Hoffman, "The Problem of Extracting the Knowledge of Experts", The AI Magazine, Vol. 8, No. 2, Summer 1987, pp. 53-67.

[Hoffman, 1989] Robert R. Hoffman, "A survey of methods for eliciting the knowledge of experts", SIGART Newsletter, April 1989, Number 108, Knowledge Acquisition Special Issue.

[Hofstadter, 1981-1] Douglas R. Hofstadter, Godel, Escher, Bach: An Eternal Golden Braid, Basic Books Inc, 1979.

[Hofstadter, 1981-2] Douglas R. Hofstadter, The Mind's I - Fantasies and Reflections on Self and Soul, Basic Books, Inc., 1981.

[Hofstadter, 1985] Douglas R. Hofstadter, Metamagical Themas - Questing for the Essence of Mind and Pattern, Basic Books, Inc., 1985.

[Howard, 1987] C. Howard and D. Rehak, "KADBASE -- A Prototype Expert System-Database Interface for Integrated CAE Environment", AAAI, Seattle, 1987.

[Jafar, 1990] Musa Jafar and A. Terry Bahill, "Validator, A Tool for Verifying and Validating Computer Based Expert Systems", in Operations Research and Artificial Intelligence: The Integration of Problem-Solving Strategies, 1990, pp. 372-385.

[Jo, 1989] Jinmyun Jo, Residual Stress Measurement in Railroad Car Wheels, Ph.D dissertation, Materials Engineering Science, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, (September 1989).

[Johson, 1988] R. A. Johson and D. W. Wichern, Applied multivariable statistical analysis, 2nd ed Prentice Hall: Englewood Cliffs, NJ (1988).

[Khanna, 1990] Tarun Khanna, Foundations of Neural Networks, Addisson-Wesley Publishing Company.

[Kohonen, 1987] Teuvo Kohonen, "Adaptive, associative, and self organizing functions in neural computing", in Applied Optics, Dec 1987, Vol. 26, No. 23, pp. 4910-4918.

[Kohonen, 1988] Teuvo Kohonen, "The Neural Phonetic Typewriter", in Computer, March 1988, pp. 11-22.

[Klimasauskas, 1988] Casimir C. Casey Klimasauskas, "Neural Networks: A short Course", in PC AI, Nov/Dec 1988.

[Kunz, 1984] J. C. Kunz, T. P. Kehler and M. D. Williams, "Applications Development Using a Hybrid AI Development System", The AI Magazine, Fall 1984, pp. 41-54.

[Kurzweil, 1990] Raymond Kurzweil, The Age of Intelligent Machines, MIT Press, 1990.

[Lafferty, 1990] Larry Lafferty, Albert M. Koller, Jr., Greg Taylor, Robin Schumann and Randy Evans, "Techniques for Capturing Expert Knowledge: an expert systems/hypertext approach", in SPIE Vol. 1293 Applications of Artificial Intelligence VIII, 1990, pp. 181-191.

[Lampert, 1989] Robin Lampert, David Hamilton and Elliot Soloway, "A case Study of Software Testers", in AI EDAM, 1989, 3 (1), pp. 35-51.

[Laufmann, 1990] Steven C. Lauffmann, D. Michael DeVaney and Mark A. Whiting, "A Methodology for Evaluating Potential KBS Applications", in IEEE Expert, Dec 1990, pp. 43-61.

[Leung, 1990] K. S. Leung and M. H. Wong, "An Expert System Shell Using Structured Knowledge", in Computer, March 1990, pp. 38-47.

[Levitt, 1987] R. Levitt and J. Kunz, "Using Artificial Intelligence Techniques to Support Project Management", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 1, No. 1, 1987, pp. 3-24.

[Liebowitz, 1989-1] Dr. Jay Liebowitz, "If There is Artificial Intelligence, Is There Such Thing as Artificial Stupidity ?", in SIGART newsletter, July 1989, No. 109, pp. 26-28.

[Liebowitz, 1989-2] Jay Liebowitz, "Helping Students Develop their First Expert System Prototype", in Interface: The Computer Education Quarterly, Summer 1989, Vol. 11, issue 2, pp. 63-65.

[Mahalingam, 1985] Sriram Mahalingam and Deva D. Sharma, "Weldex -- An Expert System for Non-Destructive Testing of Welds", in IEEE 1985 Second Conference On Artificial Intelligence Applications, pp. 572-576.

[Mettrey, 1987] W. Mettrey, "An Assessment of Tools for Building Large KB Systems.", The AI Magazine, Vol. 8, No. 4, Winter 1987, pp. 81-89.

[Mettrey, 1991] W. Mettrey, "A comparative Evaluation of Expert System Tools", in Computer, pp. 19-31, Feb 1991.

[Microsoft, 1990] Microsoft Excel, User's Guide, Function Reference, 1990.

[Mirzai, 1990] A. R. Mirzai, Artificial Intelligence - Concepts and Applications in Engineering, MIT Press, 1990.

[Mittal, 1985] S. Mittal and C. Dym, "Knowledge Elicitation from Multiple Experts", The AI Magazine, Vol. 6, No. 2, Summer 1985, pp. 32-36.

[Myers, 1986] Ware Myers, "Introduction to Expert Systems", IEEE Expert, Spring 1986, page 100-109.

[Nadler, 1990] Morton Nadler, Richard W. Conners and Eric P. Smith, Pattern Recognition Engineering, An excerpt from a book to be published as a Wiley Interscience Publication.

[Neuron, 1991] Neuron Data, "Nexpert Object", Introduction Manual, Functionnal Description, Reference Manual, Database Integration Guide, User's Guide, Application Programming Interface Reference Manual, January 1991.

[Nii, 1979] H. Penny Nii and Nelleke Aiello, "AGE (Attempt to Generalize): A Knowledge Based Program for Building Knowledge-Based Programs", in Proceedings of the Sixth International Joint Conference on Artificial Intelligence, 1973, Vol. 2, pp. 645-655.

[Nii, 1986-1] H. Penny Nii, "Blackboard Systems: The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures (Part One)", The AI Magazine, Summer 1986, Vol. 7, No. 2, pp. 38-53.

[Nii, 1986-2] H. Penny Nii, "Blackboard Systems: Blackboard Application Systems, Blackboard Systems from a Knowledge Engineering Perspective Point Of View (Part Two)", The AI Magazine, Aug 1986, Vol. 7, No. 3, pp. 82-106.

[Noyan, 1987] Ismail C. Noyan and Jerome B. Cohen, Residual Stress, Measurement by Diffraction and Interpretation, Springer-Verlag, 1987.

[O'Keefe, 1987] Robert M. O'Keefe, Osman Balci and Eric P. Smith, "Validating Expert System Performance", in IEEE Expert, Winter 1987, pp. 81-90.

[Ott, 1988] Lyman Ott, An Introduction to Statistical Methods and Data Analysis, Third Edition, PWS Kent Publishing Company.

[Pao, 1989] Yoh-Han Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley Publishing Company, Inc.

[PC Week, 1990] "Microsoft puts Windows 3.0 to work", PC week, May 28, 1990, p. 57.

[pLogic, 1991] pLogic^{TM}, Advanced Statistical Pattern Recognition Software, User Guide, pLogic Knowledge Systems, 1991.

[Potet, 1990] Jac Potet, Marc Tricard and Robert W. Hendricks, "A Data Transfer Program for the TEC X-Ray Stress Analysis System", Report RSL-90-005, Sept. 1990.

[Potet, 1991] Jac Potet, Data Flow Programs for the Creation of an Expert System to Validate and Interpret the Measurements of Residual Stress by X-ray Diffraction, Projet de Fin d'Etudes, ENSAM, May 1991.

[Prereau, 1985] D. S. Prereau, "Selection of an Appropriate Domain for an Expert System", The AI Magazine, Vol. 6, No. 2, Summer 1985,, pp. 26-30.

[Prereau, 1987] D. S. Prereau, "Knowledge Acquisition in Expert System Development", The AI Magazine, Vol. 8, No. 2, Summer 1987, pp. 43-51.

[Ramamoorthy, 1988] C. V. Ramamoorthy, and P. C. Sheu, "Object Oriented Systems" in IEEE Expert, Fall 1988, pp. 9-15.

[Reddy, 1973] D. R. Reddy, L. D. Erman, R. D. Fennel, and R. B. Neely, "The Hearsay Speech Understanding System: An example of the Recognition Process", in Third International Conference on Artificial Intelligence, 1973, pp. 185-193.

[Schank, 1987] R. C. Schank, "What is AI anyway ?", The AI Magazine, Vol. 8, No. 4, Winter 1987, pp. 59-65.

[Schulman, 1990] Bob Schulman, Statistics for Research - Part I, Course Notes, VPI&SU Statistics Department, Fall 1990.

[Schulman, 1991] Bob Schulman, Statistics for Research - Part II, Course Notes, VPI&SU Statistics Department, Spring 1991.

[Searle, 1990] John R. Searle, "Minds, Brains, and Programs", in The Philosophy of Artificial Intelligence, 1990, pp. 89-104.

[Shriver, 1988] Bruce D. Shriver, "Artificial Neural Systems", in Computer, March 1988, pp. 8-9.

[Siddal, 1990] James N. Siddal "Expert Systems for Engineers", Marcel Dekker, Inc, 1990.

[Slagel, 1988] J. Slagel and M. Wick, "A Method for Evaluating Candidate Expert Systems Applications", The AI Magazine, Vol. 9, No. 4, Winter 1988, pp. 44-53.

[Smith, 1983] R. G. Smith and J. D. Baker, "The Dipmeter Advisor System - A case study in Commercial Expert System Development", in Proceedings of the Eighth International Joint Conference on Artificial Intelligence, Vol 1, pp. 122-129.

[Stefik, 1986] M. Stefik and D. G. Bobrow, "Object Oriented Programming: Themes and Variations", The AI Magazine, Vol. 6, No. 4, Winter 1986, pp. 40-62.

[Steitz, 1989] Norbert A. Streitz, Jorg Hannemann and Manfred Thuring, "From Ideas and Arguments to Hyperdocuments: Travelling through Activity Spaces", in Hypertext '89 Proceedings, 1989, pp. 343-364.

[TEC, 1985] Technology for Energy Corporation, "TEC Model 1600, X-Ray Stress Analysis System, Operation and Maintenance Manual", TEC, One Energy Center, Lexington Drive, Knoxville, Tennessee 37932.

[Thompson, 1991] Bill Thompson and Bev Thompson, "Overturning the Category Bucket", in Byte, Jan 1991, pp. 249-256.

[Tricard, 1990-1] Marc Tricard, "An Expert System for the Validation and Interpretation of X-Ray Residual Stress Data", Projet de Fin d'Etudes, ENSAM, May 1990.

[Tricard, 1990-2] Marc Tricard and Robert W. Hendricks, "An Expert System for the Validation and Interpretation of X-Ray Residual Stress Data", Advances in X-Ray Analysis, Vol. 34, pp. 601-610, 1990.

[Turing, 1950] A. M. Turing, "Can a Machine Think?", in Mind: A quarterly review of Psychology and Philosophy, Oct 1950.

[Udpa, 1987] L. Udpa and W. Lord, "A Knowledge Based Image System for Electromagnetic Nondestructive Testing", in IEEE 1987 Second Conference On Artificial Intelligence Applications, pp. 44-49.

[Unterberger, 1990] Ronald C. Unterberger, Brian P. Butz, David. G. Miller and Neil F. Palumbo, "Control Laws Expert Assistant for Rotorcraft (CLEAR), in SPIE Vol. 1293 Applications of Artificial Intelligence VIII, 1990, pp. 494-505.

[Wasserman, 1990] Anthony I. Wasserman, Peter A. Pircher, and Robert J. Muller, "The Object Oriented Structured Design Notation for Software Design Representation", in Computer, March 1990, pp. 50-63.

[Wasserman, 1989] Philip D. Wasserman, Neural Computing - Theory and Practice, Van Nostrand Reinhold.

[Wells, 1989] Tracy L. Wells, "Hypertext as a means for Knowledge Acquisition", in SIGART newsletter, April 1989, No. 108, pp. 136-138.

[Widrow, 1988] Bernard Widrow and Rodney Winter, "Neural Nets for Adaptive Filtering and Adaptive Pattern Recognition", in Computer, March 1988, pp. 25-39.

[Zornetzer, 1990] Steven F. Zornetzer, Joel L. Davis and Clifford Lau, An Introduction to Neural and Electronic Network, Academic Press Inc.

11. VITA

The author was born on January 12, 1967 in Tulle, France. He has spent a total of six years in the French West Indies windsurfing and enjoying the sun. He received his "Baccalaureat Serie C" (scientific) in June, 1984. After three years of "technological preparatory classes", in Reims and Angers, France, he was admitted to the "Ecole Nationale Superieure d'Arts et Metiers" (ENSAM), and received his engineering degree three years later in 1989. He has spent the summer of 1989 and spring 1990 working on his Projet de Fin d'Etudes at both ENSAM in Paris, France and VPI&SU in Blacksburg, Virginia. He has been a Master of Science candidate in Materials Engineering since summer 1990.