Engineering with Computers 8, 139–149 (1992)

Engineering with Computers



# The Complementary Roles of Expert Systems and Database Management Systems in a Design for Manufacture Environment

**Garth S. Miller and Prof. Jonathan S. Colton**
Laboratory for Intelligent Design Systems, George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA USA

**Abstract.** Effective product design that satisfies functional requirements and can be manufactured easily requires vast amounts of knowledge on the part of the design engineer. This paper focuses on the complementary roles of expert systems and database management systems as they relate to the Intelligent Design System (IDS) in a Design for Manufacture (DFM) environment. Each technology complements the other in its strengths and abilities. The database management system provides the capacity for handling the large amounts of data generated during the design process. The expert system provides a reasoning mechanism for identifying manufacturing violations and generating meaningful recommendations. These components work cooperatively with a CAD interface to form a unified, intelligent design environment.

An information flow analysis of the Intelligent Design System resulted in the development of three distinct classifications of information within the database: CAD data, a design catalog, and a knowledge base. The CAD data tables employ an object oriented approach to store specific information about the physical characteristics of each part being designed. The design catalog contains cost, weight, and strength characteristics of the standard parts and fasteners used within the system. The knowledge base contains rules and heuristics concerning design and manufacturing methodologies.

The placement of the expert system rules in the database represents an innovation. As a result, the expansion and updating of the materials, fasteners, standard parts, or manufacturing processes used by the Intelligent Design System is facilitated without rule modification. Overall performance of the system is further increased due to the efficient management of the knowledge base by the database management system. This allows the designer to modify the knowledge and help the system to learn without the need for a knowledge engineer.

## 1 Introduction

A major problem plaguing industry is the lack of communication between engineering and manufacturing personnel [11,16]. Within many companies, these divisions are separate. Design engineers are concerned primarily with insuring that a part meets its functional requirements. Manufacturing engineers are left with the task of producing the part as designed, or of negotiating changes in the design when it cannot. As a result, many parts undergo a redesign process, which not only results in longer lead times, but also increases product costs.

The underlying problem stems from the vast amounts of knowledge required to design a product that satisfies its functional requirements and can be manufactured easily. "There is simply too much [knowledge] involved in the design [and] manufacture . . . of products for designers to know, and be able to apply all of it meaningfully at the initial stages of design" [4]. The key to preventing costly redesign lies in catching these errors and correcting them as early as possible in the design process.

One approach to overcome these limitations and produce manufacturable designs is the concept of the Intelligent Design System (IDS). In its simplest terms, the IDS merges a CAD system, an expert system, and a database management system. Figure 1 is a schematic of the IDS system and Fig. 2 shows the IDS interface. References [2,3,18,19] describe the development of the IDS interface and the DFM expert system. At the heart of the IDS is a knowledge base, which contains manufacturing and assembly guidelines in the form of rules and heuristics. As the engineer enters design information via the CAD interface, the expert system uses the knowledge base to check the object for manufacturability. If a manufacturing violation is detected, the engineer is immediately alerted so corrective action may be taken.

Research in this area has resulted in several prototype systems which incorporate portions of the IDS philosophy [2,4,6,7,8,12,13,14,18,19,20]. The IDS provides a mechanism for checking a design from a manufacturing perspective. Through the use of such a system, the engineer is able to design products that avoid difficult or impossible manufacturing sce-

*Offprint requests:* J.S. Colton, Lab. for Intelligent Design Systems, George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

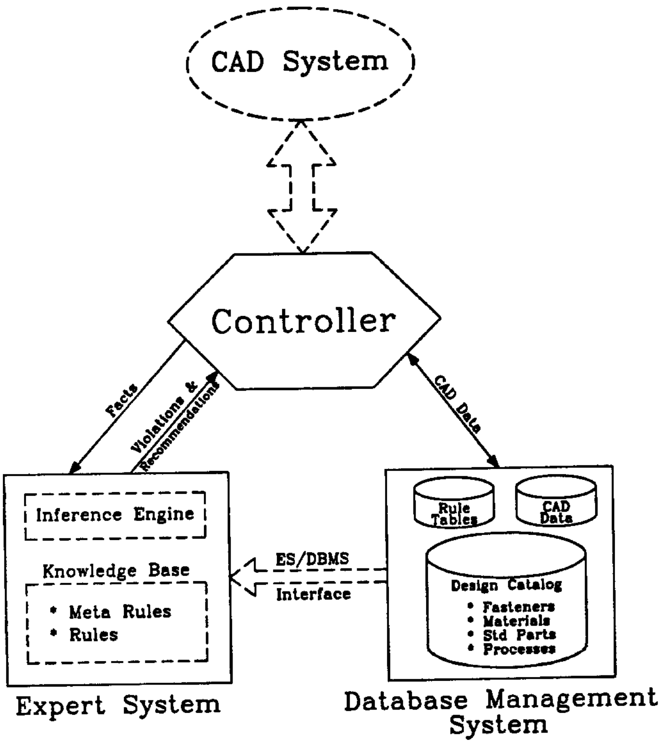



**Fig. 1.** Expert system database management system integration.

narios. The result is a manufacturable design that has bypassed lengthy and costly redesign cycles.

Examples of currently available intelligent design tools include ICAD (Intelligent CAD) [6]. It is an equation driven design tool that requires the system, or assembly, to be input as a system of equations and relationships. These mathematical descriptions form the basis for a parametric model that can intelligently adapt a design to changes in parameters. ICAD is well suited for the design of part and system "families" (e.g., heat exchangers whose type, materials, and size are functions of the environment, fluid medium, and heat transfer requirements). Pro/Engineer [12] is a similar system that allows designs to be entered in the form of equations and drawings, and then modified by varying the input parameters to produce an acceptable design. Both of these differ from the IDS in that they are based on design specific input information. The IDS contains generic information concerning design and manufacturing in general, not tied to a certain class of products.

This paper investigates the interaction between

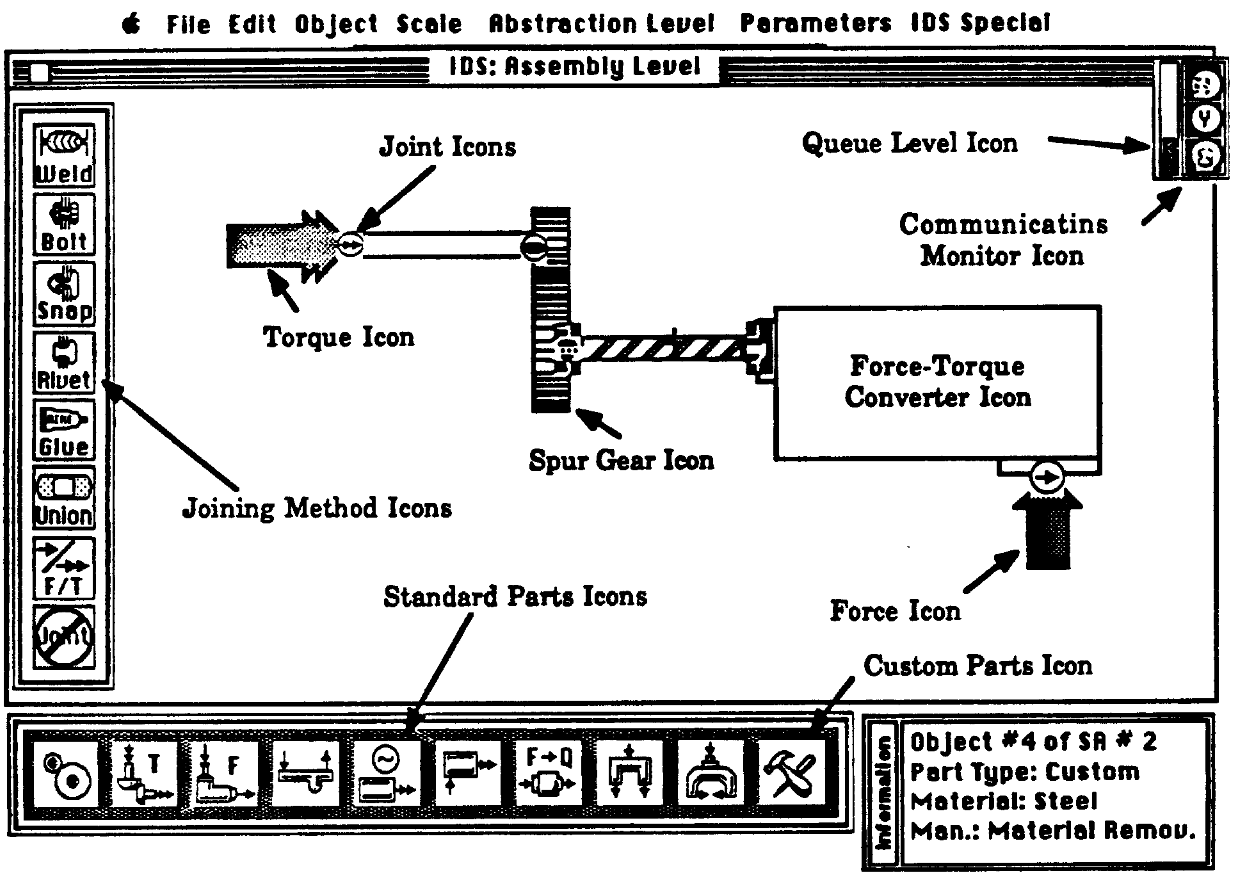



**Fig. 2.** The IDS environment screen.

expert systems and database management systems in a DFM [9] environment with a focus on merging the strengths of both technologies. The information dependencies and flow between the various components of the IDS were studied and a knowledge base consisting of relational database tables and expert system rules was developed.

## 2 Information Flow Analysis

An understanding of the flow of information in any system is vital to its successful and efficient operation. The large quantities of data controlled by the IDS intensify this need. Its information requirements may be divided into three broad categories: CAD data, a design catalog, and a knowledge base (Fig. 3).

CAD data is specific information concerning the geometric, material, and manufacturing properties of the object being designed. Traditionally, CAD data has been limited to the physical dimensions of the object: length, width, height, orientation, etc. This definition, however, is not sufficient to support the broad spectrum of information needed within the framework of the IDS. The definition used within the context of this research has been expanded to include additional information such as material properties, manufacturing processes, features [4], and associativity with other objects [18].

The design catalog is an electronic form of the vendor supply catalogs and handbooks that are an established part of the engineering design process. Like its paper counterpart, the design catalog contains cost, weight, and strength characteristics of standard parts and fasteners. In addition, physical and mechanical properties of various materials and manufacturing processes are included.

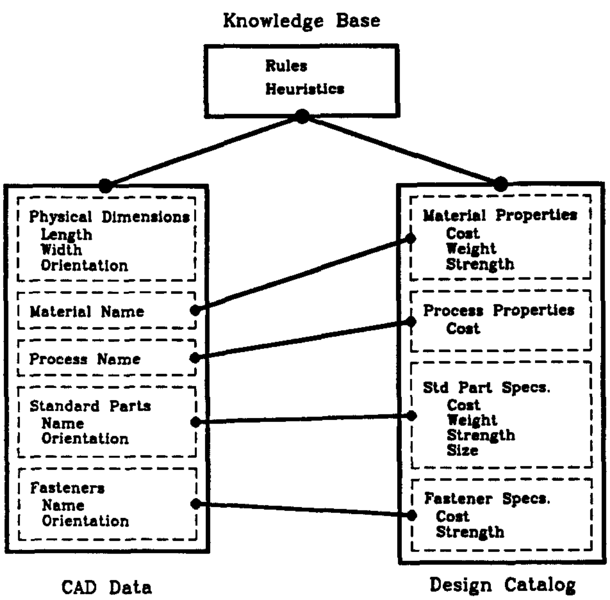



**Fig. 3.** Information flows.

The third type of information in the IDS is the knowledge base. It consists of rules and heuristics concerned with design and manufacturing methodologies. It is the rules that form the real expertise or intelligence in the IDS. Rules are used by the expert system to check the integrity of a designed object to ensure manufacturability.

The three categories of information in the IDS are interdependent (Fig. 3). CAD data is generated interactively throughout the design process. Only the minimum amount of data necessary to describe the object is stored within the CAD database. This includes physical dimensions, such as height or width, in addition to material or manufacturing process identifiers. Other detailed information about the object, such as cost, weight, and mechanical properties, is available through the design catalog. The knowledge base must also have ready access to any information in either the CAD database or in the design catalog to ascertain the manufacturability of the evolving design. The sharing of information in this manner greatly reduces data redundancies and increases efficiency.

Many complex relations exist in the DFM environment. For example, to determine the strength of a joint between two parts, it is necessary to know both the type of fastener selected and the material of each part. This is because, in many cases, the limiting factor will be the strength of the part–material combination, not of the fastener itself. Complex interactions such as these, when incorporated in an expert system, result in many lengthy, perplexing rules. A better approach is to develop a table of the material–fastener combinations (Table 1). The expert system could then, in essence, use a single generic rule to extract the strength values of the specific material–fastener combination at the appropriate time, in a similar manner to a query on a database. An example of a rule is as follows: if the steel and aluminum are being fastened with a bolt, then the strength of the joint is 1000 lb in tension.

In order to understand how to best merge expert system and database management system technologies, the respective strengths and weaknesses of each must first be examined. Database management systems (DBMS) are extremely good at managing large amounts of data, much better than expert systems alone. CAD data, design catalog, and specific

**Table 1.** Material–fastener table

| Fastener | Material 1 | Material 2 | Max tension | Max torsion |
|---|---|---|---|---|
| Nut/Bolt | Aluminum | Aluminum | 1000 | 1000 |
| Nut/Bolt | Aluminum | Steel | 1000 | 1000 |
| Nut/Bolt | Steel | Steel | 1000 | 1000 |
| Nut/Bolt | Aluminum | Plastic | 100 | 100 |
| Nut/Bolt | Steel | Plastic | 100 | 100 |
| Nut/Bolt | Plastic | Plastic | 100 | 100 |
| Adhesive | Steel | Steel | 500 | 500 |
| Adhesive | Steel | Aluminum | 450 | 450 |
| Adhesive | Steel | Plastic | 400 | 400 |
| Adhesive | Aluminum | Plastic | 350 | 350 |
| Adhesive | Aluminum | Aluminum | 480 | 480 |
| Adhesive | Plastic | Plastic | 300 | 300 |
| Weld | Steel | Steel | 1400 | 1400 |
| Weld | Aluminum | Aluminum | 1000 | 1000 |
| Weld | Plastic | Plastic | 350 | 350 |
| Snap | Steel | Steel | 250 | 250 |
| Snap | Aluminum | Aluminum | 200 | 200 |
| Snap | Steel | Aluminum | 200 | 200 |
| Snap | Steel | Plastic | 50 | 50 |
| Snap | Aluminum | Plastic | 50 | 50 |
| Snap | Plastic | Plastic | 50 | 50 |
| Rivet | Steel | Steel | 1100 | 1100 |
| Rivet | Steel | Aluminum | 1100 | 1100 |
| Rivet | Aluminum | Aluminum | 1100 | 1100 |
| Rivet | Plastic | Plastic | 300 | 300 |

rule information are best handled by the DBMS. The strength of the expert system lies in its ability to solve problems and mimic human reasoning. The simplified, generic rules of the knowledge base are located within the expert system. The expert system is able to retrieve additional rule information and other data directly from the database.

Merging the strengths of the DBMS and the expert system has several advantages over traditional methods employed in DFM analysis. First, the process of updating or incorporating additional materials, standard parts, and fasteners into the Intelligent Design System is greatly simplified. This information only needs to be added to the appropriate relational table(s) within the DBMS and can be performed by the designer without the aid of a knowledge engineer. As generic rules are used, rule modifications are not required within the knowledge base, thus eliminating a common source of error [8]. This will be discussed in section 5.

Another advantage of this approach is the reduced number of rules in the expert system. The use of the DBMS permits complex interactions between the various components of the design to be represented in a table, as discussed previously. In many cases, a single generic rule can be used to access this table of information in a concise, efficient manner. The alternative method would require one rule for each entry in the table. This will be discussed more fully in section 5.

## 3 Database Modeling and Development

The primary objective of data modeling is to identify correctly the interdependencies of data within the system domain. Poor database design usually occurs when the real data structures are hastily modeled or are not understood completely. This research uses dependency lists and diagrams to identify these interdependencies and assist in the development of fully normalized relational tables [15]. The development of a rigorous dependency list and diagram facilitates the identification and revision of problems early in the process, while they can be resolved easily.

Relational tables are composed directly from the dependency diagram. The process involves "working" single-valued and multivalued dependencies into tables by tracing the paths of dependencies through the diagram. This methodology simplifies the task of producing fully normalized tables over techniques previously available [15]. This procedure is necessary to achieve tables that are free from insertion, deletion, and update anomalies, which would severely affect the performance and integrity of the database.

An object oriented approach was adopted as the basis for modeling the CAD data. These are shown in Fig. 4 with solid lines. A complete description of all the tables can be found in Ref. [9]. One can see the large amount of data and its interdependence, even with a simple system such as the IDS. An object has been defined within the engineering design domain as a structure that describes both form and function [17]. An object is not limited to geometric information, as in traditional CAD systems. Instead, an object contains additional attribute information necessary to fully describe the part. Information such as material, manufacturing process, and features are important attributes contained within the object hierarchy. The object oriented approach further facilitates the association of parts into assemblies and subassemblies.

The design catalog contains detailed physical and mechanical properties for the various materials, processes, fasteners, and standard parts used within the IDS. This is shown in Fig. 4 with dotted lines. The various catalog components are linked to the appropriate CAD object by an identifying num-

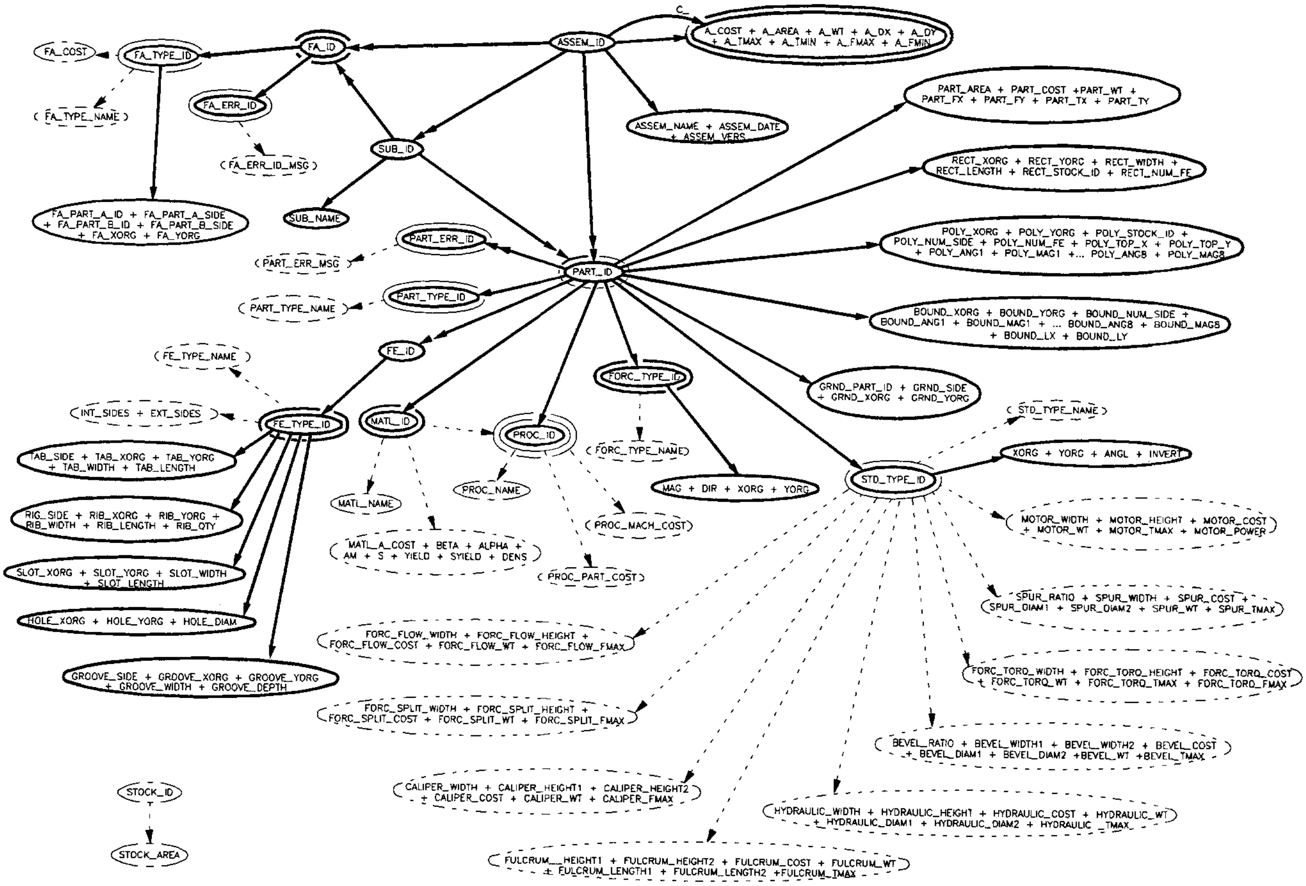



**Fig. 4.** Data dependency diagram.

ber or name. For example, the cost, density, and strength of a material can be obtained by using the material name. This association increases the efficiency of the database while reducing data redundancies.

Also contained in the design catalog section of the database are error message tables. The creation of standard error messages simplifies the task of providing meaningful feedback to manufacturing violations. Error messages are easily accessed by the expert system, thus providing an efficient, central error handling facility.

## 4 Expert System Development

Expert systems have traditionally relied on frame based or rule based methodologies for their knowledge representation. Frame based methods use a network of nodes connected by relations and organized into a hierarchy [20]. Nodes low in the hierarchy automatically inherit properties of higher level nodes. These methods provide a natural, efficient way to represent problems that are hierarchical in nature, such as locating mineral deposits or diagnosing medical diseases.

Rule based representation involves the use of "if–then" statements. Matching the "if" portion of the rule to the known facts causes the actions on the "then" side to be executed in a process called inference. Rules provide a natural way to represent processes in a rapidly changing environment [20].

This research explored an alternative method for knowledge representation. The method employed by the IDS combines concepts from both rule based and frame based methodologies. No frames are used within the expert system of the IDS. Rather, a software interface permits simple, generic rules within the expert system to access transparently frame type data and rule information stored in the relational database acting as an intelligent query on a database. The resulting combination provides increased efficiency to the overall operation of the IDS.

The use of an external knowledge base within the IDS permits the expert system to exploit directly

Question:

**Can WELD fastener be used with BEVEL gear?**

Fastener/Standard Part Table

| Fastener | Std Part |
|---|---|
| Nut/Bolt | Spur |
| Adhesive | Spur |
| Weld | Spur |
| Snap | Spur |
| Rivet | Spur |
| Nut/Bolt | Bevel |
| Adhesive | Bevel |
| Snap | Bevel |
| Rivet | Bevel |

Violation/Recommendation:

Selected **FASTENER can't be used with selected gear.**
Permissible fasteners include:
Nut/Bolt
Adhesive
Snap
Rivet

**Fig. 5.** Constituent table usage.

the power and flexibility of the relational database. Each knowledge base table contains several columns of data that represent complex interdependencies between various processes. Each row in one of these tables is essentially a rule in the traditional sense. The relational database makes it possible to use complex queries on several columns within a table to efficiently and easily extract information with a single generic rule. A single generic rule literally replaces a table full of rules, which would be required with a more traditional approach.

The knowledge base consists of three distinct classifications of tables: constituent, evaluative, and equation. The purpose of a constituent table is to determine if a given set of parameters is an allowable combination. For example, whether welding can be used as a fastening method in conjunction with a bevel gear (Fig. 5). Further, the constituent table can be used to generate recommendations in the case of a violation. In the above example, an error message would indicate that the weld fastener/bevel gear combination is not allowable and then the system would proceed to list fasteners which are acceptable for use with this fastener/gear combination: nuts and bolts, adhesive, snaps, and rivets. The system could be expanded to include suggestions as to alternative materials or manufacturing processes. This was not done here, as it was deemed not necessary to prove out the system's concepts.

Constituent tables contain from two to four search keys, for convenience. The table is searched against the given facts for a match on all keys. If a match is found, the desired operation is deemed to be allowable and no further action is required. If a match is not found, an error condition is generated. The appropriate error message and recommendation are then retrieved from the respective tables and returned to the engineer. An example is shown in Fig. 5 and is described in the previous paragraph. Constituent tables permit complex interactions between several variables to be easily evaluated where traditional algorithms may be inordinately complicated.

The second type of knowledge base table is the evaluative table (Table 1). Evaluative tables closely resemble constituent tables in their appearance and performance. The difference is that evaluative tables return data to the expert system when a match is found; for example, the strength of a joint when aluminum and steel are fastened with an adhesive. This data is returned in the form of a fact—the fastener strength is 450 lb in tension—which can be used by the expert system during inferencing. Error messages and recommendations are generated as appropriate.

The last type of knowledge base table is the equations table (Table 2). As with the previous knowledge base tables, several search keys are used as selection criteria. The equations are in the form of "greater than" and "less than" inequalities which set limits on the allowable maximum or minimum value for the associated variable. When the appropriate search keys are matched, the corresponding equations, usually several, will be extracted as a set of facts. A generic rule evaluates each new fact in the set for possible violations. Appropriate error messages and recommendations are generated as required, as shown in Fig. 5.

## 5 The Intelligent Design System Implementation

The IDS represents the merging of three separate technologies: CAD systems, expert systems, and database management systems. Commercially available applications were used where possible to minimize development time. The CLIPS [5] expert system shell and Oracle [1] database management system are two of these. The CAD system was developed at the Laboratory for Intelligent Design Systems at the Georgia Institute of Technology because of difficulties in tying into commercial CAD

**Table 2.** Equations table

| Feature | Process | Material | *Equation | Type | Err# |
|---|---|---|---|---|---|
| Hole | Deformation | any | DIAM < 1 | reqd | 2552 |
| Hole | Deformation | any | DIAM < 1.25 | pref | 2553 |
| Hole | Deformation | any | EDGE < .5 | reqd | 2554 |
| Hole | Deformation | any | EDGE < .75 | pref | 2555 |
| Hole | Deformation | any | WALL < .75 | reqd | 2556 |
| Hole | Deformation | any | WALL < 1 | pref | 2557 |
| Slot | Deformation | any | WIDTH < .3 | reqd | 2559 |
| Slot | Deformation | any | WALL < .3 | reqd | 2556 |
| Slot | Deformation | any | EDGE < .3 | reqd | 2554 |
| Slot | Deformation | any | LWRAT > 12 | reqd | 2558 |
| Hole | Solidification | Alum | DIAM < .6 | reqd | 2552 |
| Hole | Solidification | Alum | DIAM < .8 | pref | 2553 |
| any | Solidification | Alum | WALL < .87 | reqd | 2556 |
| any | Solidification | Alum | WALL < 1.2 | pref | 2557 |
| any | Solidification | Alum | EDGE < .87 | reqd | 2554 |
| any | Solidification | Alum | EDGE < 1.2 | pref | 2555 |
| Hole | Solidification | Steel | DIAM < .9 | reqd | 2552 |
| Hole | Solidification | Steel | DIAM < 1.2 | pref | 2553 |
| any | Solidification | Steel | WALL < .6 | reqd | 2556 |
| any | Solidification | Steel | WALL < .8 | pref | 2557 |
| any | Solidification | Steel | EDGE < .6 | reqd | 2554 |
| any | Solidification | Steel | EDGE < .8 | pref | 2555 |
| Hole | Solidification | Plastic | DIAM < .6 | reqd | 2552 |
| Hole | Solidification | Plastic | DIAM < .8 | pref | 2553 |
| any | Solidification | Plastic | WALL < 1.5 | reqd | 2556 |
| any | Solidification | Plastic | WALL < 1.75 | pref | 2557 |
| any | Solidification | Plastic | EDGE < 1.5 | reqd | 2554 |
| any | Solidification | Plastic | EDGE < 1.75 | pref | 2555 |
| Hole | Matl Removal | any | DIAM < .1 | reqd | 2552 |
| Hole | Matl Removal | any | DIAM > 6 | reqd | 2560 |
| Hole | Matl Removal | any | WALL < 1 | reqd | 2556 |
| Hole | Matl Removal | any | WALL < 1.5 | pref | 2557 |
| Hole | Matl Removal | any | EDGE < .75 | reqd | 2554 |
| Hole | Matl Removal | any | EDGE < 1 | pref | 2555 |

* Equations represent violation conditions and are "fired" only when a violation exists.

systems available at the inception of this research [3]. The C programming language was chosen as a basis for all code developed in conjunction with the Intelligent Design System.

The Intelligent Design System operates under a simulated distributed processing environment using two Macintosh computers. The CAD environment resides on one computer while the expert system and database manager reside on the other. A software controller monitors the information being passed between the computers (Fig. 6). A queueing system permits the engineer to continue to enter design data while the expert system performs its analysis.

A software interface was written to facilitate the extraction of data from the database by the expert system. The interface consists of a set of new expert system functions, which have been defined and added to the existing expert system commands. Each new function is actually a C program with embedded calls, which utilize the database manager. The new functions permit the expert system to access directly information within the database in a transparent fashion, thus providing the needed functionality.

Four types of interface functions were developed to provide the needed functionality of the expert system/database interface. These included general functions to *get*, *check*, and *delete* information from the database, and the ability to *write* information to the database.

Get functions are used to retrieve specific information from the database. Get functions were written in a generic fashion to allow access to any one of a number of tables with multiple search keys and to return several variables. For example, to find the cost of a specific fastener, the get function requires the name of the table, the column name, the fastener type, and the name of the fact to be returned. Get functions allow the expert system to retrieve information directly from the database that would otherwise be stored in the frames of a more traditional expert system.

Check functions provide a functionality not readily available in the traditional expert system. Check functions are used to search a table for a match on a given set of constraints. If the desired match is not found, then the check function reapplies the search with one less constraint variable in order to generate a recommended list of alternatives. For example, if the materials to be fastened by welding are plastic and steel, the check function will determine that welding is not permissible and generate a list of acceptable fastener alternatives for the selected material combination.

Delete and write functions are used to update error tables and other informational fields as calculated by the expert system. All other updates are performed directly by the database manager.

Should a manufacturing violation be found, the engineer will be notified immediately so that corrective action may be taken (Fig. 7). As can be seen from the figure, the engineer has designed a part on the system, and then placed a hole too close to an edge. The system has discovered the error and informed the designer of his mistake. In the IDS, error messages are advisory in nature. If the designer chooses to make the desired changes, then the database is marked and the correct ANSI cross hatching for the material is placed on the part. The designer can choose to ignore the messages, in which case the database is so marked and the correct ANSI cross hatching is not placed on the part in question. This serves as a visual cue to the designer that the expert system thinks that he has made a mistake. In both cases, the designer can continue to work; the system does not stop him until he takes corrective

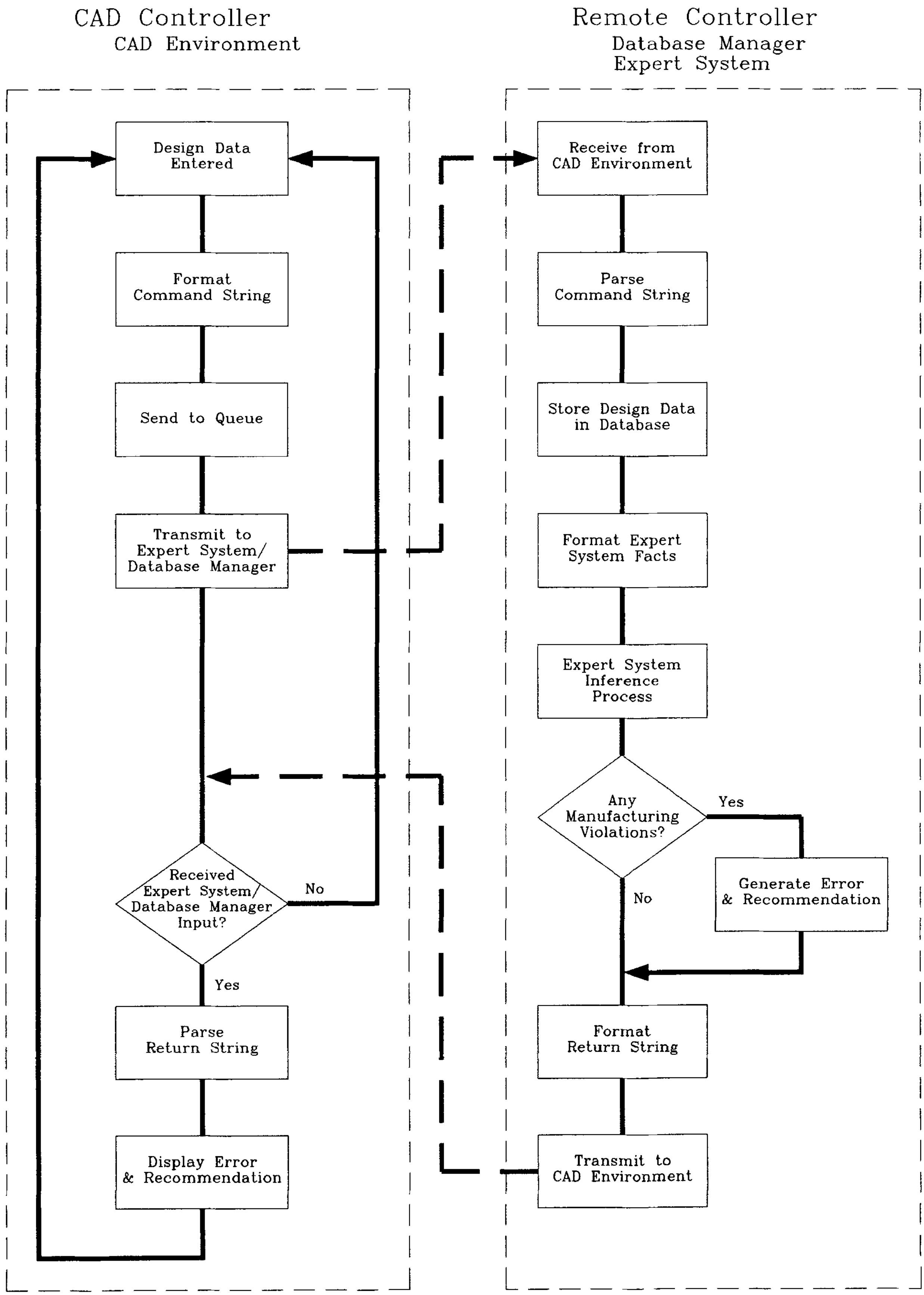
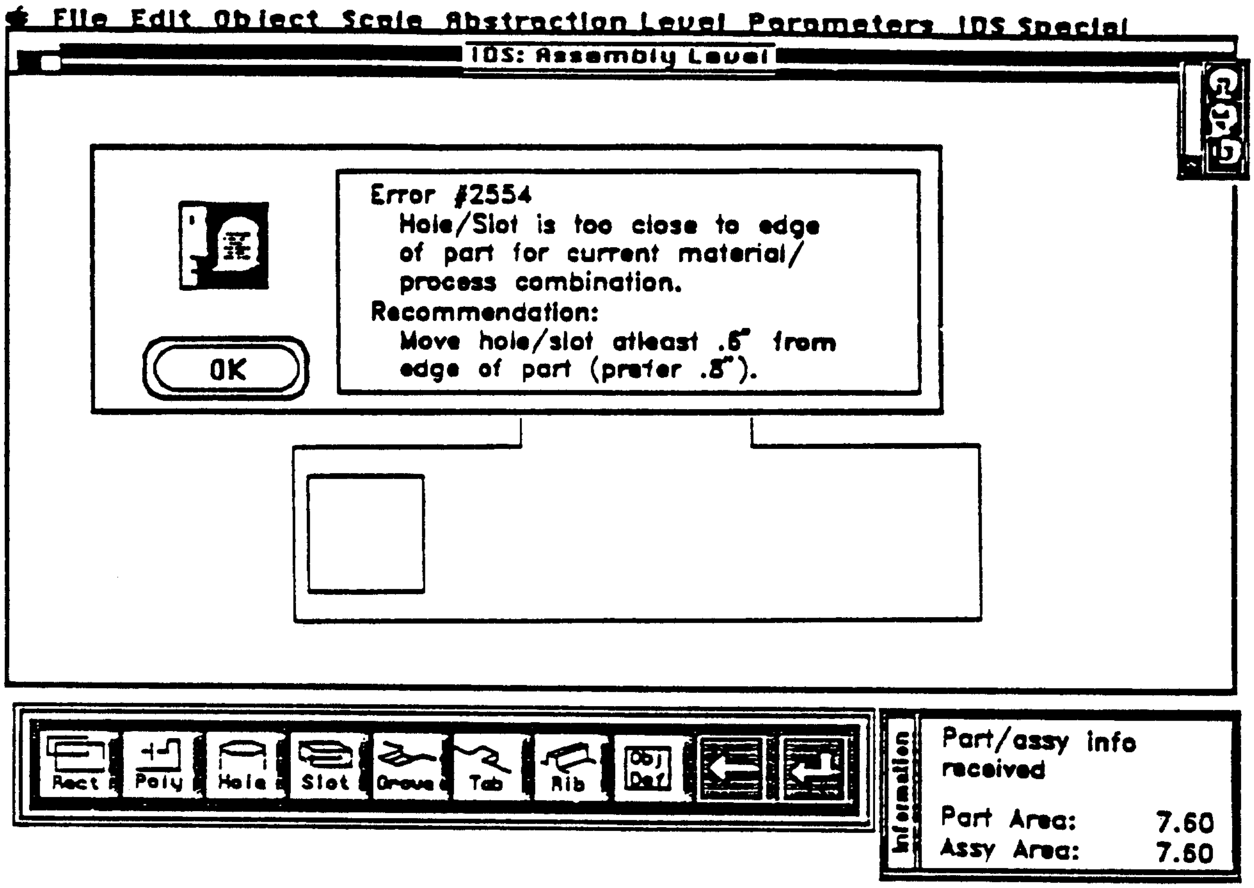



**Fig. 6.** Intelligent design system communications flowchart.




A slot is cut in the custom part. The physical dimensions of the slot are sent to the expert system.

Command String:

25 56 1 2 1 1 4 67 30 40

| | |
|---|---|
| 25 | Command Code, add feature |
| 56 | Assembly Identification |
| 1 | Part Identification |
| 2 | Part Type Classification |
| 1 | Feature Identification |
| 1 | Feature Type Classification, slot |
| 4,67 | Origin |
| 30 | Width |
| 40 | Length |

The expert system detects a violation in the placement of the slot. The appropriate error message is retrieved from the error message table, and a recommendation is generated.

Return String:

13 1 7.60 7.60 1 !2554 "Hole/Slot is too close to edge of part for current material/ process combination. Move hole/slot atleast .6" from edge of part (prefer .8")

| | |
|---|---|
| 13 | Return Code, feature added |
| 1 | Part Identification |
| 7.60 | Part Area |
| 7.60 | Assembly Area |
| 1 | Status Code |
| !2554... | Error Message/Recommendation |

Error message and recommendation are displayed in the CAD error window. The crosshatching of the part is removed, indicating an unresolved manufacturing violation.

**Fig. 7.** Adding a feature.

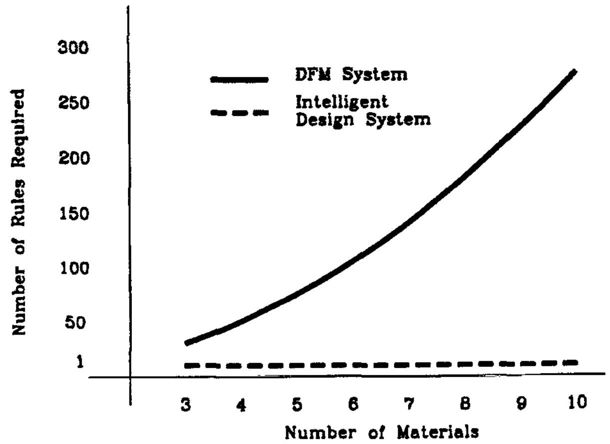



**Fig. 8.** Effect of increasing materials on rules.

action. The IDS philosophy is that the designer has the last word. This feedback to the designer is educational in value. A novice designer will be instructed as how to produce a design that is manufacturable. An expert designer can be instructed in the latest techniques and methodologies through modifications of the rules and hence the system's feedback.

The placement of the majority of the expert system rules in a database represents an innovation. This greatly reduces the number of rules that must be written. As can be seen in Fig. 8, the amount of rules that needs to be created in a traditional expert system increases dramatically if one desires to add materials, fasteners, or processes. Only the database needs to be expanded in the IDS. As a result, the IDS has a quicker response because less rules need to be fired for each test of the design. Expansion of the database does not noticeably effect the system's response because correctly formulated queries of a database are relatively insensitive to its size. As a result of this technique, more efficient and flexible knowledge based systems can be designed, constructed, and used. In a traditional expert system, rules or frames would need to be modified or added if materials, fasteners, or processes were changed or added.

This philosophy can be used in the factory to aid in process planning (CAPP). Suppose that there are a number of machine tools of varying ages and conditions in a factory. If the machines all belong to one class, for example, hydraulic presses, then one generic set of rules could be used to describe the performance of this family within the expert system and used to infer the best machine to use for each operation from the available machines. The database could hold the specific information for each of the machines, such as tolerances and availabilities. As they age, it would be a simple matter to update the information in the database to reflect the changing operational characteristics of each one. This would provide a flexible system to be developed and maintained.

This system is advisory in nature; it presents possibilities. It does not optimize or make a selection based on heuristics. The latter could be easily added and in the future may. The philosophy of the IDS is to let the designer choose.

## 6 Conclusions

The merging of expert system and database management technologies marks a significant achievement in the ability to represent the complexities of the DFM environment. Each technology complements the other in its strengths and abilities. The database management system is capable of handling the large amounts of data generated through the mechanical design process. The expert system provides a reasoning mechanism capable of identifying manufacturing violations and providing meaningful recommendations.

The storage of portions of the knowledge base in the database provides opportunities to exploit the power and flexibility of the database manager. The storage of rules in the database permits complex problems with highly interdependent variables to be solved easily where algorithmic solutions fall short. Generic rules facilitate the retrieval of data and equations from a relational table when search criteria are met.

The storage of rules in the knowledge base provides a number of benefits. First, the number of rules within the expert system is greatly reduced. Less rules means the expert system operates more efficiently and provides timely and meaningful feedback to the engineer. Development and debugging time of the expert system is also reduced due to the decreased number of rules.

Even more benefits can be seen if the system is to be expanded or updated. Addition of materials, fasteners, standard parts, and manufacturing processes is facilitated by adding records to the appropriate relational tables; no rules need to be created or modified and an expert is not needed. The resulting time saved by this method in adding even a single fastener or material is substantial and errors are significantly reduced.

The system also has an educational value; it can help instruct novice designers as well as experts.

## Acknowledgments

The research reported on in this paper has been sponsored in part by grants from the Ford Motor Company and the National Science Foundation (DDM 8957848).

## References

1. Cronin, D.J. (1988) Mastering ORACLE Featuring ORACLE'S SQL Standard. Indianapolis, Hayden Books
2. Dascanio, J.L.; Colton, J.S. (1991) An integrated intelligent design environment. Engineering with Computers, 7, 11–22
3. Dascanio, J.L. (1989) The Development of an Integrated Intelligent Design Environment. MS Thesis, School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, July
4. Dixon, J.R. (1988) Design with features: building manufacturing knowledge into more intelligent CAD systems. Proceedings of the Manufacturing International '88, ASME, Atlanta, March, 1, 51–57
5. Giarratano, J.C. (1989) CLIPS User's Guide. Artificial Intelligence Section, Lyndon B. Johnson Space Center, June
6. ICAD (Intelligent CAD). ICAD, Inc. Cambridge, MA
7. Jakiela, M.J.; Papalambros, P.Y. (1989) Design and implementation of a prototype 'intelligent' CAD System. Journal of Mechanisms, Transmissions, and Automation in Design, 111, 252–258
8. Mayer, R. (1985) Artificial intelligence applied to manufacturing. CIM Review, Fall, 25–29
9. Miller, G.S. (1990) The Complementary Roles of Expert Systems and Database Management Systems in a Design for Manufacture Environment. MS Thesis, School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, March
10. Ohsuga, S. (1989) Toward intelligent CAD systems. Computer-Aided Design, 21:5, 315–323
11. Poli, R.; Graves, R.J.; Sunderland, J.E. (1988) Computer-aided product design for economical manufacture. Proceedings of the Manufacturing International '88, ASME, Atlanta, March, 1, 23–27
12. Pro/Engineer, Parametric Technology Corp., Waltham, MA
13. Rogers, J.L. (1989) A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem. NASA Technical Paper 2903
14. Roger, J.L.; Feyock, S.; Sobieszczanski-Sobieski, J. (1988) STRUTEX—a prototype knowledge-based system for initially configuring a structure to support point loads in two dimensions. Artificial Intelligence in Engineering Design, edited by J.S. Gero. Boston: Computational Mechanics Publications, 315–335
15. Smith, H.C. (1985) Database design: composing fully normalized tables from a rigorous dependency diagram. Communications of the ACM, 28:8, 826–838
16. Starkey, J.M.; Florin, G.J. (1986) Design for Manufacturability. ASME paper 86-DET-121
17. Ullman, D.G.; Dietterich, T.G. (1988) Progress in understanding the process of mechanical design. Design Theory '88, Proceedings of the 1988 NSF Grantee Workshop on Design Theory and Methodology, Amherst, MA, June, 1.1.1–1.1.11
18. Wade, J.L.; Colton, J.S. (1989) A framework for feature-based representation of the design process. Proceedings of the Computers in Engineering Conference: Engineering Database Management: Leadership Key for the 90's, ASME, Anaheim, CA, July, 97–102
19. Wade, J.L.; Colton, J.S. (1990) The development of a design for manufacture expert system. Proceedings of the Manufacturing International '90, ASME, Atlanta, March, 2, 69–75
20. Waterman, D.A. (1986) A Guide to Expert Systems. Reading, MA: Addison-Wesley Publishing Company