6-1-1989

A Knowledge-Based System for the Evaluation of Beam-to-Column Connections

Marcello Barone

Keith J. Werkman

Donald J. Hillman

John L. Wilson

ADVANCED TECHNOLOGY FOR LARGE STRUCTURAL SYSTEMS

Lehigh University

A KNOWLEDGE-BASED SYSTEM FOR THE EVALUATION OF BEAM-TO-COLUMN CONNECTIONS

by

Marcello Barone
Keith J. Werkman
Donald J. Hillman
John L. Wilson

ATLSS Report No. 89-11

June 1989

An NSF Sponsored Engineering Research Center

A Knowledge-Based System for the Evaluation of Beam-to-Column Connections

ABSTRACT

The Designer Fabricator Interpreter (DFI) system is a knowledge-based computer tool that attempts to bridge the information interface gap between design engineers and fabricators of structural steel systems. The current prototype DFI system, acting as an intelligent interface between the designers and fabricators, focuses on critiquing designs of steel beam-to-column connections in buildings based on standard steel fabrication and field erection procedures. The DFI system incorporates fabricator and erector heuristics in the form of rules in an object-oriented frame-based knowledge representation scheme which hierarchically models beams, columns and the component pieces of beam-to-column connections in buildings.

1. INTRODUCTION

The Designer Fabricator Interpreter (DFI) project is part of a comprehensive research effort intended to provide an environment which fosters communication between various parties (agents), i.e. owner, architect, designer, fabricator and erector, involved in a construction project. DFI currently contains a thin slice of knowledge to evaluate the geometric fit-up of beam-to-column connections from a fabrication and erection point of view. The system gives structural designers the ability to check their preliminary conceptual connection design against general fabrication and erection knowledge to determine how their initial design decisions may effect the overall downstream fabrication and erection processes.

In this report, present industrial practices are described in Section 1. Section 2 describes the functioning of DFI and also provides an illustrative example. The technical aspects of DFI are presented in Section 3. Section 4 summarizes the current research efforts while in Section 5 a plan of future research is given.

1.1. Present Practice

Fragmentation in the U.S. construction industry has caused a decline in its ability to compete successfully on a global scale. The industry is made up of many small and large privately owned companies, each with its own distinct construction procedures and information flow practices. Due to the industry's fragmented nature, the most current engineering information is seldom used in the field. Typically, as problems arise in the field, the contractors make notes on their drawings. At times, the drawings in the field are as many as three revisions behind the engineer's most current drawings. This common practice does not allow for vital engineering information to be communicated in a timely fashion between agents of the construction process [1]. This, at times, results in construction delays and cost overruns. Since there are many agents involved in the construction process, there is a need for intelligent interfaces to provide various perspectives on agent viewpoints and clear explanations to individual end-users.

1.2. The DFI Approach

The Designer Fabricator Interpreter system is an initial attempt at developing a framework for cooperative problem solving between construction agents, specifically designers and fabricators. Toward this end, research has been focused on obtaining and formalizing design and fabrication knowledge pertaining to beam-to-column connections in buildings. This research includes the identification of design and fabrication processes, modeling of agents' beliefs and the determination of both unique and shareable knowledge aspects of design and fabrication operations.

The expected end user of the DFI system will be the practicing structural design engineer. The system is intended to behave as a *standby (surrogate) advisor* providing additional design and fabrication expertise to aid the user in predicting potential downstream problems with his initial proposed design.

2. HOW THE SYSTEM WORKS

The menu-driven DFI system has two stages of operation. The first stage involves user entry of building data from data files and system checking of consistency of data in those files. The data include a basic description of the building, including beam schedules, column schedules, and framing plan. The system then evaluates the consistency of these data using any required external databases such as the American Institute for Steel Construction (AISC) database of shape parameters. The user is given explanations for any problems that are found and suggestions on how to correct the data.

The second stage of the system allows the user to interactively enter a connection and then study DFI's critique of that connection. Through a series of brief menus, DFI prompts the user for the location of the beam-to-column connection and all other necessary information such as connection rigidity, connection detail material, e.g. top flange angle, bottom flange plate, and connection fasteners, e.g. shop welded, field bolted. Once the connection input is complete, it is evaluated and critiqued by DFI. The critiquing process utilizes parameters taken from the AISC database to perform calculations to determine if physical fit-up is possible. These calculated results are then used in the fabrication and erection heuristics. For example, if framing angles are shop attached to the column flange then the bottom flange of the beam must be coped to allow for erection of the connection. If inconsistencies in entered data are found or impracticalities in fabrication are discovered, the user may determine the source of the problem by reviewing the trace of the rule tree. This trace may be examined either from graphical output or textual output. The DFI system provides detailed explanations and suggestions of the evaluated rules where appropriate.

The process described above is summarized in the DFI Information Flow Diagram shown in Figure 2-1. Table 2-1 describes the information flow lines in Figure 2-1. Following this, Table 2-2 provides a description of the modules in the figure.

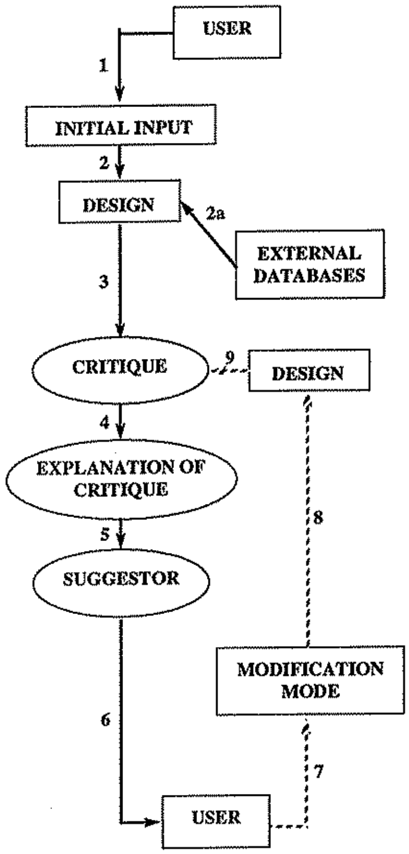

Figure 2-1: DFI Information Flow Diagram

Table 2-1: Description of DFI Flow Lines

FLOW LINE	DESCRIPTION OF FLOW LINES
1.	User inputs the initial design data (beam and column schedules, framing plan and proposed connection detail).
2.	DFI internally represents the design and produces a graphical display for design verification by the user.
3.	The design is critiqued by DFI.
4.	An explanation of the critique is presented.
5.	Suggestions are generated based on the critique.
6.	Suggestions are presented to the user.

At this point, the user chooses the most appropriate suggestion for his particular situation. If a modification is necessary the system must be restarted and the entire process repeated.

DFI EXTENSIONS:

7.	User chooses from a menu of suggestions and makes any necessary modifications.
8.&9.	Modifications are incorporated, and the new design is evaluated

2.1. Illustrative Example

In Figure 2-2 an endplate connection is described. After the user has loaded in his building data, he then selects a floor on which the specific connection resides. After selecting the floor, a simplified floor plan is presented in DFI's upper graphic window which shows only the beams that frame into the columns. The user then selects a specific beam-to-column connection which is then displayed in the lower window.

The user is then prompted for the type of connection (endplate or flange and web). The system prompts for type of fastening method (bolting or welding) for both the column and beam ends of the connection material. In this example, an endplate connection is selected with the following parameters. The column fastener is field bolted and the beam fastener is shop welded. These fastener methods are considered standard for an endplate connection.

Even though the endplate is a standard connection, the DFI system points-out that the connection has potential fabrication and erection issues that the design engineer should consider. Specifically, the system points out that *"An endplate is very unforgiving to*

Table 2-2: Description of DFI Modules

INITIAL INPUT

- User enters a data file name

- System then loads :

1. Column Schedule
2. Beam Schedule
3. Framing Plans

DESIGNER INPUT

- User specifies connection using menu driven input

- Required Input :

1. Location of connection
2. Rigidity of the connection
3. Type of connection (i.e. Endplate)
4. Detail material (i.e. Top Flange Angle)
5. Fastener types (i.e. Shop Weld, Field Bolt)

CRITIQUE

- System evaluates all input using the rule hierarchy

- There is no feed back to the user at this point

EXPLANATION

- Once the rule evaluation is complete the user may trace the rule hierarchy to see how the critique was done.

- Detailed explanations of the rule evaluation are presented to the user during the menu driven rule trace.

SUGGESTOR

- During the rule trace, suggestions are presented to the user

- The suggestions accompany the detailed explanations

- All suggestions presented are not prioritized. Therefore, the user must choose the most appropriate suggestion for his particular situation

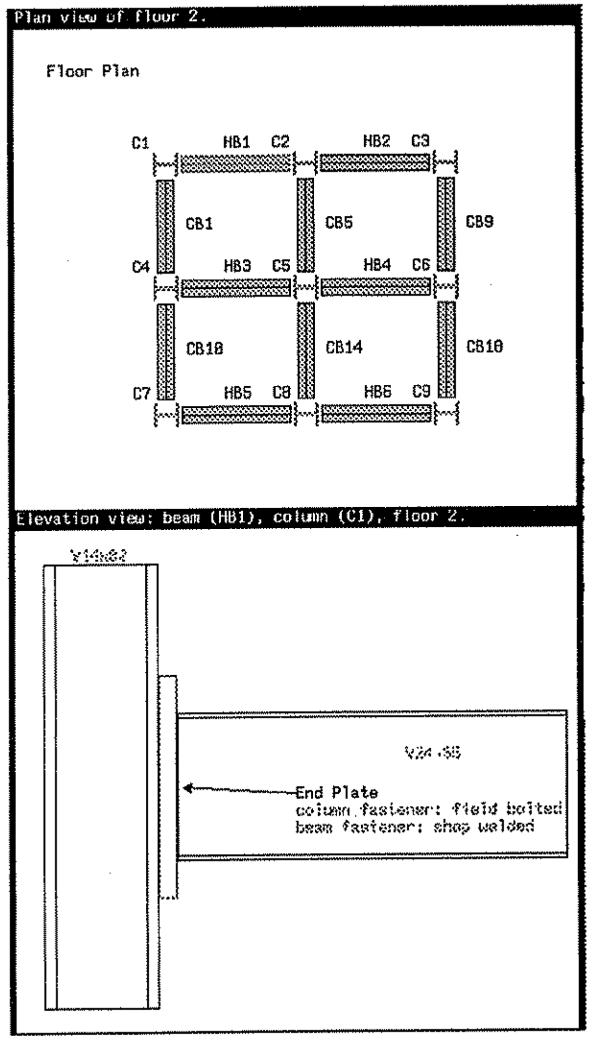

Figure 2-2: DFI Window Display

variations in beam length." The system suggests that the an alternate connection method may be easier for this type of connection or to specify tolerances to insure that connection will fit in the field. In this case, the actual rules that fired for this evaluation are displayed as shown in the upper left window in Figure 2-3. The textual explanations are displayed in the large window at the right.

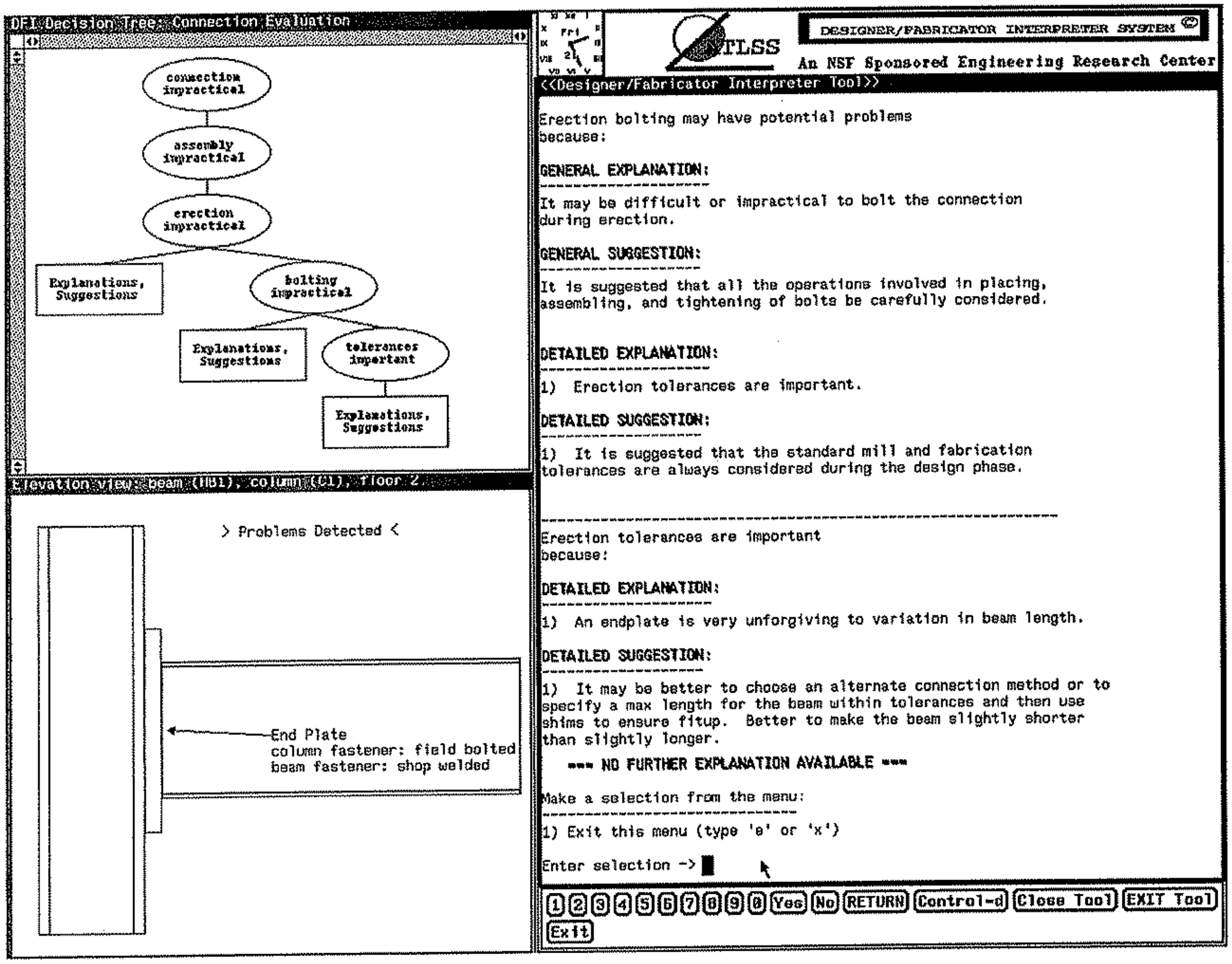

Figure 2-3: Endplate Connection Evaluation

3. TECHNICAL ASPECTS

The prototype version of the DFI system provides a critique of the designer's proposed connection from the viewpoint of practical and economical fabrication and erection. To accomplish this, the DFI system incorporates fabricator and erector heuristics in the form of rules in an object-oriented frame-based knowledge representation scheme which hierarchically models beams, columns and the component pieces of beam-to-column connections in a hierarchical fashion.

3.1. System Components

The system is composed of various software modules including a representation scheme for the component pieces of a connection in a building, a graphical and menu-based user interface and a backward-chaining inferencing mechanism utilizing object-oriented goal-based rules. These modules are briefly described in the following subsections. The DFI prototype is implemented in Quintus Prolog using Quintus ProWindows for its graphical interface. The system currently runs under the Sunview windowing environment on Sun workstations.

3.1.1. DFI knowledge representation.

The knowledge representation scheme is centered around a frame-based [2] hierarchical part, part-of representation of a building which is decomposed into a group of objects (parts) that are ordered hierarchically from a root object (a building) to component connection objects (bolts and welds). Figure 3-1 depicts the part, part-of decomposition hierarchy of a beam-to-column connection and its relation to a building. In DFI, a building is composed of column lines that are composed of column members. Floors, composed of beams, intersect the building's column lines. The intersection of a floor with a column line is represented as a connection node and can have up to four beams framing into the column at 90 degree angles. Each beam framing into a column has its own unique instance of a beam-to-column connection object. This connection object is further decomposed into a column, beam and connection materials, such as endplates or flange and web connections.

3.1.2. DFI user interface.

The frame-based subsystem also provides control for generating input menus and output graphics. Menu prompting is provided by procedures which are attached to frame slots. When the value of an object frame's slot is requested, the associated slot procedure will return either a menu of choices or a default value. The user either selects a menu item or enters his own value. Thus, the type of information entered dynamically determines the sequence of menu prompting. Verification of user input is provided by graphics which display the connection and its component pieces.

The system also generates graphical output dynamically from both user input and internal inferencing. All graphical items are objects associated with the Prolog-based graphical interface. These graphical objects, being similar to DFI's frame-based connection objects, are tightly coupled with the connection component objects which they graphically represent. DFI's graphical output, appearing in several windows, includes a floor plan of the building at specified floors, an elevation view of the specific user selected connection and a decision tree of the actual rules that fired during the connection evaluation.

3.1.3. DFI inference mechanism.

As a result of a connection evaluation, a decision tree is generated from the rule interpreter's application of goal-based rules to the data in the building hierarchy. The rules are sectioned into three rule sets, each applied at a specific time in a given context during system's operation. Each rule set consists of a decision tree based on a single goal which is then decomposed into a series of subgoals that are represented as rules and subrules in a hierarchical fashion. The rule inferencing process that best suited these goal-based rules was a backward-chaining, fully exhaustive methodology. Thus, rules composed of disjunctive subrules will have all of their "OR" subrules evaluated regardless of the truth of each subrule. This could result in a rule possibly having several disjunctive subrules supporting it. This form of inferencing is done to assure all impractical conditions (fabrication and erection oversights) are identified and presented to the user for review.

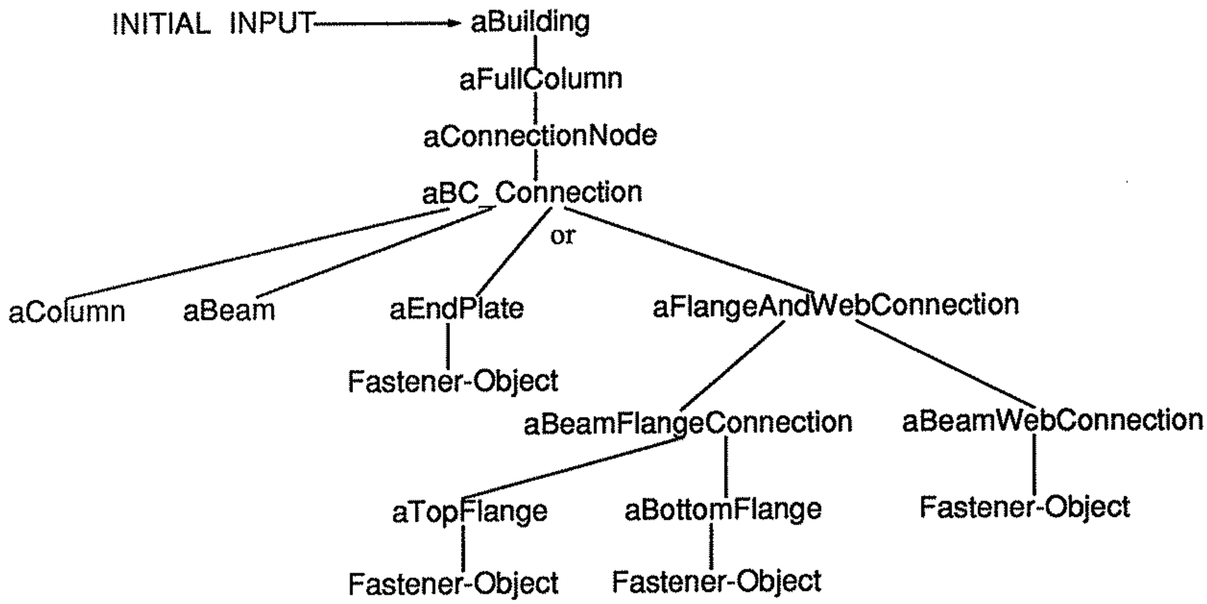

Figure 3-1: Frame Hierarchy of DFI

The hierarchical, goal-based DFI rule structure also allows for easier rule maintenance due to a dependency ordering resulting in the decomposition of rules (goals) into subrules (subgoals). In a flat rule structure, the inferencing of rules may lead to conflicts due to the lack of inference ordering. This may require the need for adding additional control information which usually appears within each rule. This is not desirable since, as new rules are entered, a ripple effect may result, and several rules may need to be rewritten, simply because control information may have to be changed. Also, a hierarchically structured rule set permits rules to be written more easily [3] because people tend to describe tasks in terms of decomposed subtask hierarchies.

4. SUMMARY

A pilot prototype of the Designer Fabricator Interpreter (DFI), a knowledge-based system for the evaluation of the geometric fit-up of right angled beam-to-column connections, has been developed. This system provides structural designers the ability to critique proposed connection designs, and to determine what potential fabrication or erection problems that could be encountered downstream in the construction process. The purpose of the pilot prototype was to provide a test bed for knowledge and data representations related to beam-to-column connections in buildings. By dealing with a thin slice of the domain, researchers were able to show a working system (i.e. a proof of concept) within a few months.

5. CURRENT RESEARCH

Current DFI research involves the development of an intelligent connection design environment for engineers. Such a system will build upon the current critiquing capabilities of DFI during the connection input phase to intelligently guide the user to a standard connection design based on common fabrication practice. Also, a set of alternative connection configurations will be generated from the user's input and any other known data. The user will then be able to evaluate the entered connection along with the alternatives to explore the downstream implications such as economics, feasibility and ease of assembly of the evaluated connections. Thus, models of the viewpoints of each agent's (design, fabrication and erection) beliefs (conceptual schemes) [4] must be developed to evaluate the proposed connections intelligently.

In addition to developing agent models, a scheme must be devised for the problem-solving interaction between cooperative agents. The current scheme in DFI includes design, fabrication and erection agents, each with his own viewpoint (set of beliefs) for evaluating a connection. Initially, the design agent will analyze the connection and try to determine the intent of the engineer's design, selecting "good" characteristics of the connection for later use in suggesting alternatives to the engineer. During this process, the design agent will query the fabrication and erection agents for their viewpoints on the values of the specified characteristics. It is quite possible that a characteristic which the design agent determines to be beneficial in the final connection design is detrimental to the fabricator and erector because of increased labor costs.

In order to provide for a cooperative evaluation of connection characteristics, the system will need to represent shareable knowledge that is common to all agents as well as unique knowledge contained within each agent. To this end, an open systems model has been developed [5] which allows for the incorporation of diverse bodies of knowledge without extensive reformatting of the knowledge. This model is being extended to include a knowledge sharing scheme which is shown in Figure 5-1. In this example, the design agent posts a query based on economics. Both the fabrication and erector agents generate a response to the query in terms of their be using their own unique knowledge of their processes. During the query process, knowledge transformations will be used to reconstruct fabrication and erection answers for the design agent. This process of cooperative problem solving involves agents reasoning about other agents' beliefs or viewpoints. The models of agent expertise and knowledge transformations are under development.

One goal for DFI is to expand the current knowledge base to include additional heuristic fabrication/erection knowledge as well as design knowledge. A computer architecture is being developed where common agent (designer/fabricator/erector) knowledge is stored in a *knowledge core* and agent specific knowledge is held within separate agent *knowledge bases*. After primary evaluation of the connection, by DFI, additional specific designer, fabricator or erector agents will be identified to provide a more comprehensive evaluation from their specific perspectives. This is shown in Figure 5-2 where the user has chosen Designer1, Fabricator2, Erector1 to evaluate the proposed structure.

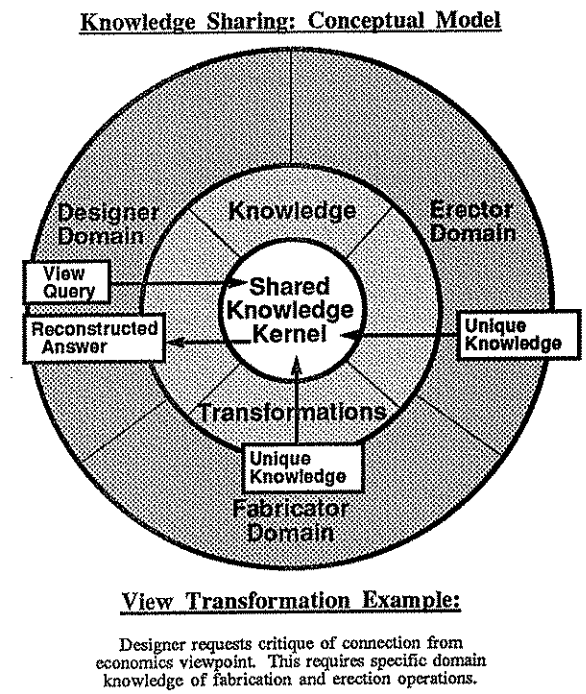

Figure 5-1: Shared Knowledge Model

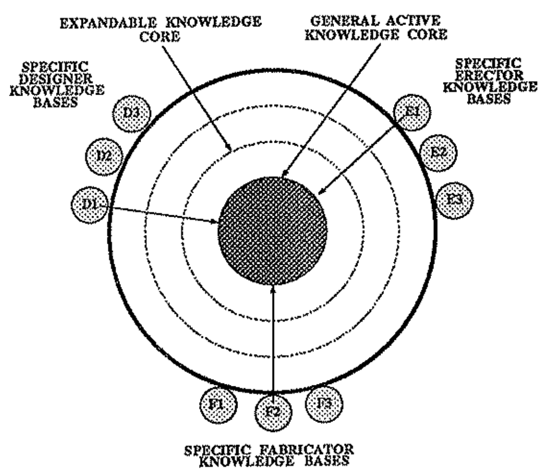

Figure 5-2: Shared Knowledge Bases

Other DFI goals include providing structural engineers assistance in designing and evaluation entire buildings through the use of an open-system model linking fabrication, erection and design knowledge-bases to a closely coupled graphical user interface. This approach could provide an environment wherein buildings would be evaluated in a more comprehensive and consistent manner prior to the award of fabrication or erection contracts. This system, as a step toward computer integrated construction, will require certain common construction practices to change. Through refinement and industry acceptance, the DFI system can lead to a reduction in design changes. More importantly it can help reduce the mismatch between the intent of the designer and the ability of the fabricator to perform economically and productively, thus resulting in an overall project cost reduction.

References

[1] Simpson, G.W. & Cochran, J.K., "An Analytic Approach to Prioritizing Construction Projects", *Civil Engineering Systems*, Vol. 4, No. 4, p.185-190(1987).

[2] Fikes, R. & Kehler, T., "The Role of Frame-Based Representation in Reasoning", *Communications of the ACM*, Vol. 28, p.904-919(1985).

[3] Winston, P.H., *Artificial Intelligence*, Reading, MA, Addison-Wesley Publishing Company (1984), 2nd edition.

[4] Allen, J., *Natural Language Understanding*, Reading, MA, Benjamin/Cummings Publishing Company, Inc. (1988), In Ch.15 "Belief Models and Speech Acts".

[5] Wong, S.T.C., "The Open-System Model for Knowledge-based Systems Development," NSF-ERC ATLSS Report 89-06, Lehigh University, Bethlehem, PA, USA (1989).