Res Eng Des (1990) 2:35–52

Research in Engineering Design

Managing Information for Concurrent Engineering: Challenges and Barriers

Steven Salzberg[1] and Michael Watkins[2]
[1]Department of Computer Science, Johns Hopkins University, Baltimore, Maryland, USA; and [2]Harvard Business School, Boston, Massachusetts, USA

Abstract. An ongoing research project with a large international manufacturing company has uncovered many critical issues for the development and introduction of systems that support *concurrent engineering*. Several of these issues can be solved through careful interface design, while others require significant technical and organizational changes. To reduce training costs and support a broad spectrum of users, computer system developers must create a single system image, giving the users the impression that they are accessing a single database through a consistent and easy to use interface. This interface, and the tools it contains, may be used to support concurrent engineering activities even when a company is geographically distributed over a large area. The complexity of the firm's products and processes and the need for rapid access to relevant performance information make it essential that developers identify suitable frameworks for organizing database queries. *Design hierarchies*, representations of the structure and function of the firm's products and processes, have proven to be powerful tools for effective query management, and for efficient navigation through the database. The database interface must also produce integrated displays of data drawn from a number of sources in response to prestructured queries. Beyond these interface design issues, there are a number of technical and organizational barriers to the implementation of large-scale engineering systems. In particular, the existence of many incompatible databases in different parts of an organization makes the introduction of a new, uniform system very difficult. Organizational issues also play a major role in achieving, or hindering, the implementation of new computer systems. This paper describes some of the technical innovations, and the motivations behind them, from one particular engineering design system. It also discusses the reactions of engineers and management, and explains why management may oppose innovation even when engineers enthusiastically support it.

1 Introduction

Many engineering groups within a manufacturing company can benefit from easy access to design, cost, and performance data on the firm's products and processes. Because most new product and process designs are variants of old designs, product design engineers can use comparative data on the cost and performance of the firm's and competitor's products in their analysis. As part of a *concurrent engineering* (or simultaneous engineering) process, knowledge of the performance limitations of the firm's manufacturing processes helps product engineers design products that are easier to manufacture. Manufacturing engineers can use information on the field performance of products to alert them to problems that are influenced by manufacturing process design. cost, process capability, and process reliability data for existing processes can be used by manufacturing engineers to make appropriate selections of new equipment (e.g., to improve process capability by improving machine tolerances). If engineers responsible for plant operations can view data on problems with manufacturing lines in other plants, then they can distinguish between plant-specific problems and process design problems. These are but a few of the many advantages of an integrated concurrent engineering system.

Ideally, all of this engineering information would be kept in one large database, and any engineer, anywhere in the company (which might have operations in widely scattered locations, even different countries or continents) could quickly get access to it. One of the main purposes of our research was to determine if such a system could be created today. We discovered that no large manufacturing company in the world has such a system. A second discovery was that implementation of such a system is much more difficult than we had hoped.

We outline herein our approach to developing an engineering information system for a large manufac-

---

Offprint requests: Prof. Steven Salzberg, Department of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA turing company. We begin by outlining our research goals and methods, and then present detailed examples from two prototype systems developed as part of our effort to implement concurrent engineering through the use of new computer systems. We conclude with a discussion of problems for such systems, including both technical issues and organizational barriers. While our proposed system addresses and solves some of these problems, others clearly remain open. We hope to make database designers more aware of the practical problems confronting those who wish to implement engineering databases, so that future databases may address some of these issues. Our approach is a combination of theory and experiment, in that we followed our studies of concurrent engineering with system-building efforts. The resulting prototype versions of our solutions allowed us to obtain better information about the tools required for concurrent engineering.

2 Research Goals and Methods

The goal of our research project was to investigate the role of information technology in problem-solving activities within manufacturing organizations. Recent comparative studies of design lead time differences among manufacturing firms have pointed to differences in organizational capabilities for problem solving. The degree of overlap between the problem-solving activities of product engineers and those of process engineers has emerged as a key determinant of performance differences among firms [Clark and Fujimoto 1989]. However, much of the existing research on the effectiveness of engineering problem-solving has focused on differences in project organization, and problem-solving methods, not on the impact of information technology.

We chose to proceed by exploring, in careful detail, the use of information technology in the engineering organization of one large manufacturing firm. We carried out our research at a company that we will call Auto Technology Inc. (ATI), in order to preserve confidentiality. ATI produces automobiles, probably the world's most complex mass-produced engineered products. They produce a small number of distinct models in very high volumes, with many different options available for each model. Product and process design activities are distributed among several locations in three countries. Plant operations and dealer networks are distributed around the world. Because of the wide distribution of personnel and activities, many engineering activities must be conducted in parallel. Such parallel work on a single product is what is meant by the term "concurrent" or "simultaneous" engineering.

ATI was a good site for our study because senior management was already very concerned about the performance of the product and process design organizations. Recent vehicle programs had been plagued by a large number of problems late in the design process and early in manufacturing start-up. Internal studies had convinced management that their design organizations were not competitive with Japanese companies on any of the critical dimensions of design quality, design lead-time, or design productivity[1]. In design lead-time alone, external studies show a gap of at least a year between ATI and its Japanese competitors [Clark et al. 1987, Clark and Fujimoto 1987]. At the same time, the high reliability and "user friendliness" of Japanese products had raised the level of customer expectations of design quality. Management believed that design quality, productivity, lead time, manufacturing cost, and manufacturing quality all had to be improved significantly.

ATI is a complex organization with a multitude of existing manual and computer systems that influence product and process engineering. In such a dynamic and complex environment, we knew that we needed a lens to focus our attention on critical issues. We therefore decided to concentrate on systems that support the improvement of design quality. After a detailed examination of existing information systems that influenced design quality, we chose to develop new information systems and use them as research instruments. We believed that a system development approach to research would offer deeper insights into the management of the design process: insights that would be complementary to those obtained by studies of organizational structure and processes. We controlled the design of prototype systems and could observe organizational responses to these systems. In effect, the systems that we constructed became experiments that led to a conceptual framework for the design of engineering support systems.

3 Initial Investigations

We began our research at ATI by looking into ways to automate an engineering analysis technique called Failure Modes and Effects Analysis

---

[1] ATI had recently obtained an interest in a Japanese company, and ATI personnel had carried out extensive comparative studies of the Japanese firm's engineering and operations activities.

(FMEA). FMEA is a technique that was developed by the US aerospace industry to insure that complex systems were thoroughly debugged [e.g., Dussalt 1983]. The FMEA approach is quite straightforward: an engineer carries out a structured review of his design prior to its release for manufacturing. He considers every known way that something could fail in the design, and generates a list of these failure modes. For each failure mode, he then writes down the effects and root causes of the failure, and the actions he has taken to prevent the failure. Finally, he makes assessments of the severity and probability of each failure mode. He combines these assessments into a "criticality" number that determines how much design effort will be devoted to each failure mode. The intent of FMEA is to force the engineer to think carefully about his design before it goes into production, avoiding costly redesign efforts or more disastrous results when the product is in the field. An ideal FMEA results in a design that has been completely verified on paper before a single physical product is manufactured.

Unfortunately, FMEA has not worked very well in ATI. In part this is due to ATI's approach to engineering. FMEA is a design tool, but many engineers tend to "develop" new vehicles rather than design them[2]. That is, engineers rely heavily on a series of pilot vehicles to test their designs, despite the high cost of such vehicles. The value of an FMEA in this framework is not clear to many of ATI's engineers. In addition, we found that many engineers find the FMEA process extremely tedious and time consuming—which, unfortunately, it is.

Our study of FMEA led us to investigate ATI's process for testing vehicles and manufacturing processes prior to start-up (a process they call "design verification"). Although ATI relies heavily on prototype testing, we found that such testing fails to identify a significant number of problems in ATI's products and processes. We also found that ATI's engineers and managers believed that most of these problems were reasonably foreseeable. The large volume of apparently forseeable problems has the unfortunate effect of obscuring truly novel and important problems[3].

Many of the problems with ATI's vehicles are identified in the field by the customers. This results in high warranty costs, occasional vehicle recalls, and, most critically, a lower reputation for quality. Even if problems are discovered in pilot testing, it is often too late for a complete resolution. Late design changes tend to be patches that may themselves result in new problems in the plants or the field. We concluded that ATI could benefit from a much greater emphasis on analysis in the design process. This analysis, which we called "theoretical testing," was needed to supplement physical testing.

Although we found widespread dissatisfaction with FMEA and ATI's existing system for design verification, most engineers acknowledged the need to improve the process of debugging new designs. On the basis of our initial study, we decided not to attempt to automate the FMEA process. Instead, we decided to investigate the engineering process at ATI in more detail, and to develop computer support for engineering analysis and design verification activities at ATI.

We next investigated the most common kinds of design problems in ATI's products and processes. We looked at how these problems were recognized, diagnosed, and solved. We organized groups of product, process, and plant engineers to study the design of a cross section of vehicle systems: the tailgate assembly, the transmission output shaft assembly, the pedal box, the engine camshaft, and the water pump. In addition, we discussed the sources of engineering problems with a selection of engineering managers at different levels in the ATI hierarchy.

3.1 Inaccessible Information

We found that most of ATI's problems with products and processes were not due to oversights or mistakes by engineers. Given access to appropriate information, engineers usually did a credible job, and sometimes an excellent one. Problems arose when information was inadequate or in an unusable form. In many cases, engineers failed to use pertinent information—even though it was available within the company—because they did not know it was available, or because the cost of obtaining the information and translating it into a useful form was too high. For example, one vehicle had design problems that remained unresolved for several years because the engineers never received *usable* feedback from the customers. Most feedback came from customer surveys, which were performed by marketing groups, and were summarized and distributed in

---

[2] Discussions with engineers in ATI revealed that some firms, such as Austin Rover in the UK, are known to be more analytical in their approach to vehicle engineering, whereas others like ATI favor a more developmental approach.

[3] We observed this phenomenon most clearly in ATI's plants. The plant engineers had so many problems confronting them that they could only create "patches" to treat the symptoms. In such circumstances, the firm is unable to learn from novel problems.

huge, disorganized books of data that few engineers had time to examine.

We also discovered that many of the problems with ATI's products and process were similar to problems that have occurred before. Repetition of problems was exacerbated by the loss of experienced engineers through retirement, and a trend toward more frequent rotation of engineers among jobs (these features led ATI at one point to consider building an expert system to replace engineers, but that was clearly impossible). In investigating interactions between plant engineers and design engineers, we found that a large number of the "small" problems that occurred in the plants were recorded only locally on plant drawings, and not distributed to other plants or to the design engineers. Details of the capability of plant processes were kept on paper drawings and other records located in or near the plant, so that design engineers had no way of knowing in advance if their new designs would be easy or impossible to manufacture.

3.2 Functional Barriers

The most serious barriers to information flow were interfunctional: between the product design groups and the process design groups, and between the process design groups and the plant engineering organizations. In these interfaces, the usual communications problems were compounded by a divergence of goals and interests, and by geographical dispersion. The company had long recognized the importance of improving communication, and had formed many *ad hoc* groups of engineers to discuss design issues. These "simultaneous engineering" groups typically met on a monthly basis to discuss issues that crossed the usual organizational boundaries. However, logistical problems made it very difficult to operate such groups. Engineers usually had to fly to the meeting place from several different locations, and much time was spent in simply arranging the meetings. Furthermore, the groups included engineers at different levels in the company hierarchy, and the sensitivity to that hierarchy within ATI made it very difficult to have an open discussion.

It became clear to us that ATI needed a more effective way to disseminate many types of performance information—market data, cost data, design data—to many different engineers. We also concluded that a *problem* database, containing a history of problems and solutions, would dramatically reduce problem repetition, if the information could be made easily accessible to all engineers. The obvious solution was to create a database to hold all these types of information, and give engineers relatively free access to the data.

3.3 Technical Barriers

When we describe our work to computer scientists, we are often asked, "wait a minute, don't they already have a big database with all the engineering data in it, including all the types of data you are describing?" The answer is, emphatically, *no*. ATI, like most large manufacturing companies, has a large number of incompatible, often redundant databases. In retrospect, it is easy to see how this situation evolved: each time a group created a database, it used its own budget, and very often purchased new database software. Sometimes new hardware was purchased as well. The organization stipulated that any cost must be justified by expected savings *within* the user group. Communication with other systems was not required, and was therefore not a priority for most new database applications. After many years of operating in this fashion, ATI has many different database systems, all using different data formats, running on different hardware, and managed by different people. There is no central repository where one can find out if a given datum is available within any of the company's computers, and if so on which computer. Even if an engineer knows that data is available, getting it from another part of the company requires him to go through (often formal) channels, sometimes requiring significant paperwork. His perception is that the time and effort required to obtain information usually outweigh the benefits.

To make matters more difficult, a significant proportion of the data in existing databases is not useful to engineers. Since most of the existing corporate databases were designed for management control, not engineering support, the existing data is often not in a form appropriate for use by engineers.

4 Developing Experimental Systems

Once we had completed our initial investigations, we began work on our experimental engineering support systems. We decided to focus our initial efforts on supporting the design of a single component. The component that we chose was the liftgate (also called the hatchback) assembly. The liftgate was a good choice for several reasons. The customer operates the liftgate directly, so comparative performance data can be collected through customer surveys. It is a relatively discrete system of moderate complexity, and it consists of a mixture of externally supplied and internally produced components. Also, the manufacturing processes that are used to produce it are similar to a number of other systems, such as doors, hoods, and trunk lids, and some of the manufacturing processes, such as painting and panel pressing, are used for all outer body parts.

We conducted extensive discussions with both design and manufacturing engineers (all of whom had worked on liftgates) to collect additional data, and to discuss the functions that they would like to see in an "ideal" design support system. A primary issue with the engineers was easy access to the data, and in particular the need for frameworks that would organize the data. Another issue was the creation of overviews of data that had been culled (automatically) from several distinct sources. Below, we discuss these issues and others in greater detail.

4.1 Interface Design Principles

Over the years, ATI's engineers had become cynical about "engineering support systems." They had been presented with one computer system after another, each one just as difficult to learn and use as the previous one. The time that they lost in learning to use these systems was rarely justified by the benefits of the system. For example, we found that ATI management had budgeted 2 full weeks of training time for each engineer to learn to use a single recently introduced database system. Since deadline pressures did not permit engineers to devote large blocks of time to computer training, they had to spread out this training over a considerable period of time, which mad it more difficult. At the time of our study, few engineers had even begun to learn to use the system. To complicate matters, most engineers were not convinced that the system would help them anyway, in part because management had presented it as a *fait accompli,* with no opportunity for the engineers to suggest changes in format or content. We concluded that ATI could realize substantial advantages by, first of all, designing a system in close consultation with engineers, and making sure that it met their needs, and that they would be able to use it with minimal training.

We knew that we had to create an interface that was as transparent to the engineer as possible. We chose a navigational, hypertext-like approach to interface design to match the information-seeking behavior we had observed in design engineers. ATI's engineers seldom access information in a sequential, prespecified manner. Instead, they typically need a few specific chunks of information to resolve each of a number of design concerns. The requirement for information chunks tends to evolve as an engineer proceeds from an initial set of issues into related sets of issues, and it is impossible to tell at the beginning which chunks of information will be sufficient and which will lead to a need for much more information. To increase their efficiency, engineers need the capability to retrieve information that is related, for design purposes, to the information being viewed at any given time. We cannot stress strongly enough that the logic of the information interconnections must be the user's logic, not the database developer's. We will discuss the process of identifying these interconnections in greater detail later.

To build a system that satisfied the requirements of ATI's engineers, we chose a few simple interface design principles to guide us.

1. *Develop a single system image.* The interface should create the impression that the user is accessing a single database, regardless of the underlying structure of the data. Where possible, the single system image concept should be extended across all applications that the engineer uses.
2. *Use direct manipulation techniques.* To support direct manipulation [Hutchins et al. 1986], the system must have a simple control mechanism, such as a mouse, to permit the user to execute commands with a single action (e.g., pressing a button or selecting a menu item). All commands should be available on menus to simplify learning and mitigate the effects of forgetting. The system should provide visual or auditory feedback after every user action. Where possible the system should also give the user some indication when a user action would result in a system response. For example, any area of the screen which responds to a mouse click should highlight itself when the mouse moves over it.
3. *Support mode-free operation.* System commands should have consistent or similar effects throughout the system. To further support the goal of single system image, this consistency in command semantics should be extended across all the systems that engineers use. This makes the system more predictable and easier to learn.
4. *Allow easy back up and escape.* The user should never feel trapped in the system. He should always know how to escape, even if he takes a path he's never explored. For example, every screen should have a "back up" button to allow the user to return to the previous screen. In addition, all commands should be reversible.
5. *Minimize input demands on users.* Typing

should be kept to a minimum, since some engineers do not know how to type. When an engineer must enter data, the system should provide defaults for inputs that are frequently required, such as using the current date as the default for the date of occurrence of a problem. The system should carefully check such input to guarantee its validity.

6. *Maximize clarity in data presentation.* No screen should be so cluttered with information that an engineer cannot absorb it in less than 30 sec. Graphs should be used where appropriate as alternatives to standard table formats.

ATI's engineers reacted enthusiastically to the interface of our experimental systems. Most of them had never seen a system that offered them such a simple control paradigm, and many of them asked us when they could have it for their own use. One of the engineers working with us, who had little previous computer experience and poor typing skills, was operating the system within 10 min, and was demonstrating it to other engineers after 1 h of practice.

4.2 Using Design Hierarchies for Query Management

Minimizing the cost of information means more than having the database respond efficiently to queries. Database designers must also minimize the time it takes an engineer to formulate and execute a suitable query to the database. As domain models become more complex, effective query management becomes more important. One approach to the problem of query management is to provide very general and powerful tools for query generation and storage, i.e., database query languages. This approach, although suitable for experienced users of conventional database systems, is unsuitable for the vast majority of working engineers using an engineering database. For an engineering database to be broadly accepted and used, inexperienced users must be able to retrieve some genuinely useful information from the system almost immediately.

An alternative approach to the query management problem involves using domain-specific frameworks or models to organize prestructured sets of queries. By using a domain-specific model for query management, inexperienced users can derive immediate benefit from the system. Advanced users can build on the domain-specific frameworks using more general and powerful query generation facilities.

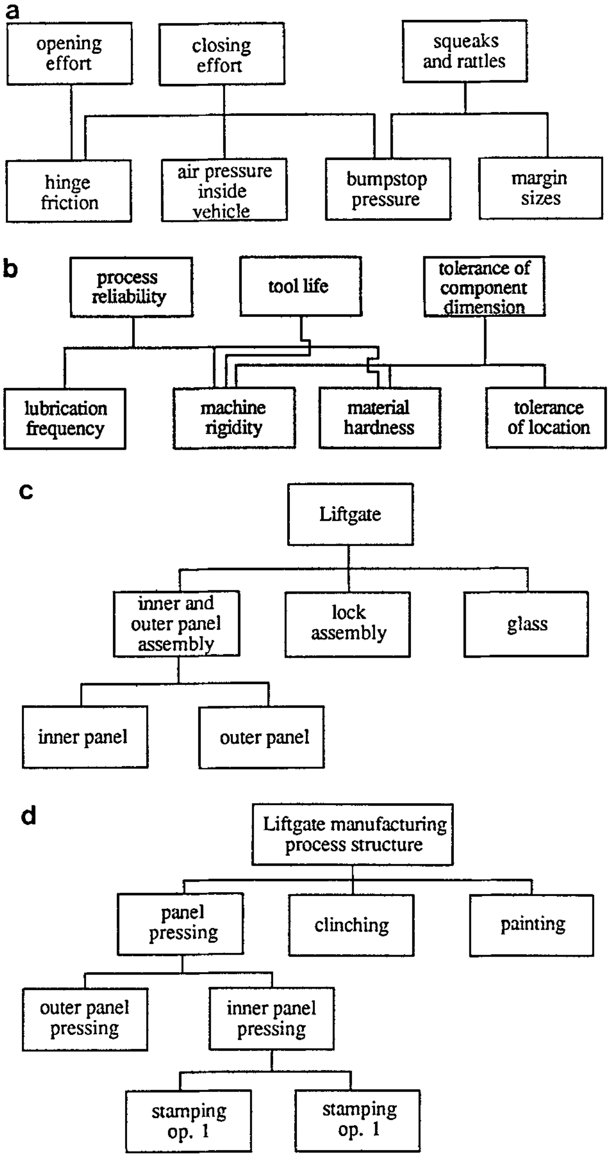


Fig. 1. Structural and functional hierarchies. **(a)** Product function hierarchy. **(b)** Process function hierarchy. **(c)** Product structure hierarchy. **(d)** Process structure hierarchy.

In our system we adopted a domain-specific approach by using *design hierarchies* to organize related sets of database queries. These design hierarchies are representations of the structure and function of ATI's products and processes. As shown in Fig. 1a and b, functional hierarchies connect the overall functions of a system (e.g., the closing effort of an automobile's liftgate) to the engineering attributes (controls, design parameters) that influence those functions. We also use structural hierarchies, shown in Fig. 1c and d, which are representations of the physical structure of a system.

The distinction between the structure and function of a system is a fundamental tool for organizing knowledge and reducing complexity in engineering and the sciences. Hierarchies are basic tools that engineers use to manage complexity and to increase their understanding of a system, and hierarchical representations are commonly used by them to represent levels of structure and function in a system [Nau 1987]. Database systems that organize information around functional and structural design hierarchies mirror an engineer's mental connections between chunks of information. Such systems appear to provide an efficient, easily comprehensible framework for the organization of design information.

In addition to the distinction between structure and function, we also distinguish between product and process hierarchies. The four types of design hierarchies that result—product function, product structure, process function, and process structure—are shown in Fig. 1. The product function hierarchy for the liftgate, of which a small piece is shown in Fig. 1a, has "customer attributes" at the top of the hierarchy. Customer attributes are functions that customers want a liftgate to fulfill. Market researchers identify these functions for each vehicle subsystem. In our system, an engineer then breaks these customer attributes down into more detailed "engineering attributes," things that engineers can actually measure and control. This breakdown continues, if necessary, down to the level of component dimensions and tolerances. Similar functional hierarchies can be developed for manufacturing processes, beginning with functional performance attributes, such as process capability and reliability, as shown in Fig. 1b. Once again, top level nodes can be broken down into more detailed engineering attributes (e.g., dimensions and access points). The breakdown of a vehicle system into subassemblies and smaller parts constitutes a product structure hierarchy, as shown in Fig. 1c. A process structure hierarchy that decomposes the manufacturing of a vehicle subsystem into its constituent processes and machine is shown in Fig. 1d.

Each type of hierarchy provides a suitable framework for organizing access to particular types of engineering data. We used product function hierarchies to manage access to product performance information, manufacturing capability information, and problem histories. Cost, weight, and failure data queries can be effectively integrated using a product structure hierarchy. Process function hierarchies can be used to organize manufacturability and maintainability information. Process structure hierarchies are suitable for machine reliability and diagnostic information.

We created several design hierarchies by working closely with ATI's engineers. Once an engineer has created the functional hierarchy for a system, such as a liftgate, he can use it, with minor modifications, to organize information for subsequent liftgates. The importance of reusability of design hierarchies is reinforced by the fact that many of ATI's designs are variants of previous designs. Design hierarchies are representations of the firm's knowledge of and experience with system design. If the information gathered during one design program is organized around core system design approaches, then engineers can use it to reduce the cost of problems in subsequent design programs. The engineering database then becomes "intellectual capital," and a source of tremendous benefits to the firm. (Some of the information in these hierarchies is also captured by the "Quality Function Deployment"—QFD—technique [Hauser and Clausing 1988], which creates tables rather than hierarchies. Hierarchies offer the advantage of capturing more than two levels of structure.)

Of course, the benefits that the firm accrues by creating explicit design hierarchies might be outweighed by uses of these hierarchies that prevented innovation (see Clark [1985] for a more detailed discussion of design hierarchies and innovation). In other words, engineers who rely on previous designs for all of their ideas will be unlikely to create truly innovative designs. Fortunately, much of the information gathered and included in the system we present below focuses on external measures of performance[4]. A clear view of the verdict of the marketplace should act to counterbalance any tendency to rely too heavily on past designs.

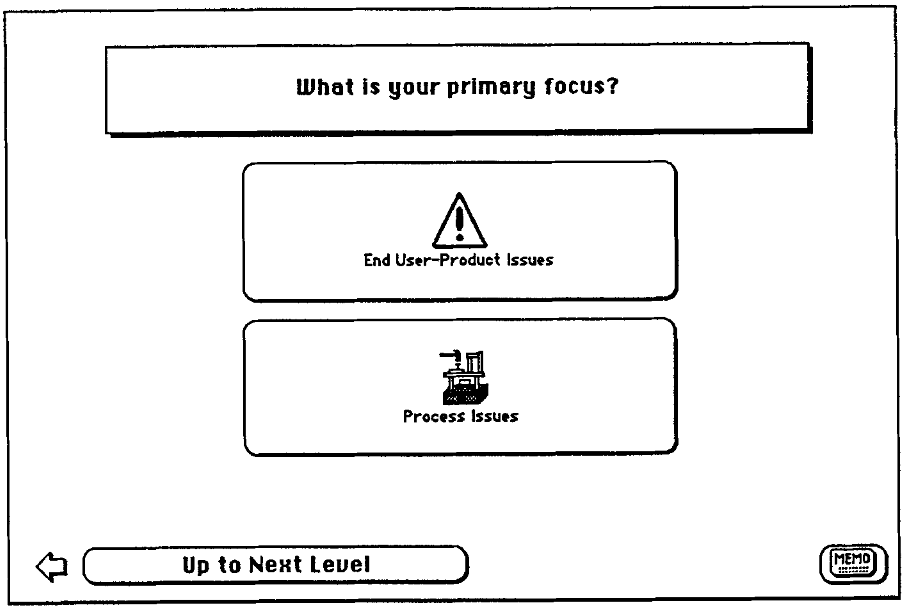
4.3 Implementation of Experimental Systems

Below we give a more detailed description of the system itself, paying particular attention to the interface. Recall that our goal was to create a system that the user could browse in a hypertext-like fashion, jumping among connected pieces of information with great flexibility.

4.3.1 First experimental system. We supported navigation through the system by placing buttons on each screen. Each button took the user to other screens with related information. Due to the complex nature of the design task, each screen in the system had many connections to other screens. We used the hierarchical organization of the informa-

---

[4] This includes market research to study customer preferences, and reverse engineering to compare process capabilities with those of competitors.


What is your primary focus?

End User–Product Issues

Process Issues

Fig. 2. One of the early screens in the interface.

tion as the base structure for navigation within the system. The figures which follow demonstrate the kinds of information contained in our first prototype system[5].

Figure 2 is one of the first screens an engineer encounters after starting up the system. The two large buttons in the middle of the screen allow the engineer to choose between the product design part of the system or the manufacturing design part. The buttons at the lower corners appear on every screen in the system. The lower left button allows the user to back up to the screen he just left. The lower right button allows the engineer to escape to a note pad, where he can jot down any comments that occur to him. The "up to next level" button also appears on many screens in the system. This button moves the user up to the next higher level in the hierarchy.

Figure 3 shows a screen that permits the user to choose one of several types of manufacturing information. Some of the underlying databases may contain text, while others contain graphs and digitized images, and they may be organized quite differently, but the interface remains uniform throughout. In Fig. 3, the manufacturing data concerns a particular process ("clinching") for automobile liftgate (also called "tailgate" or "hatchback") manufacturing.

Figure 4 shows a section of the product function hierarchy for the liftgate. In this case the nodes of the hierarchy are critical dimensions that influence system functions. Figure 4 also includes a navigation map that shows the user where he is in the hierarchy. Each of the nodes in the hierarchy is a button leading to further information, e.g., on the history of problems involving this design function.

Figure 5 shows the type of information that is obtained if the user clicks on one of the nodes in the product function hierarchy. The user would go to this screen after selecting a node called "latch pin alignment" in the product function hierarchy (this node does not appear in Fig. 4, but does appear elsewhere in the same hierarchy). The screen shown in Fig. 5 provides an English description of the critical dimension and some of the problems associated with it, plus a drawing of the components involved. The drawing was taken from a separate database of engineering drawings. In addition, there are several other buttons on the screen that take the user to other information; e.g., to process controls that might be used for latch pin alignment. The button with the icon of a miniature network takes the user back to the appropriate point in the product function hierarchy.

Figure 6 shows some of the information that can be accessed from the manufacturing part of the database. This screen is a schematic of the floor layout as a fictional plant ("Lisbon") for a particular clinching process. Each of the boxes describes one step in the process, and each box is mouse-sensitive. When the user clicks the mouse over one of these process steps, the system calls up a history of problems that have occurred in that process step in

---

[5] The basis of the interface is the Macintosh HyperCard™ system. HyperCard and MacIntosh are registered trademarks of the Apple Computer Company.

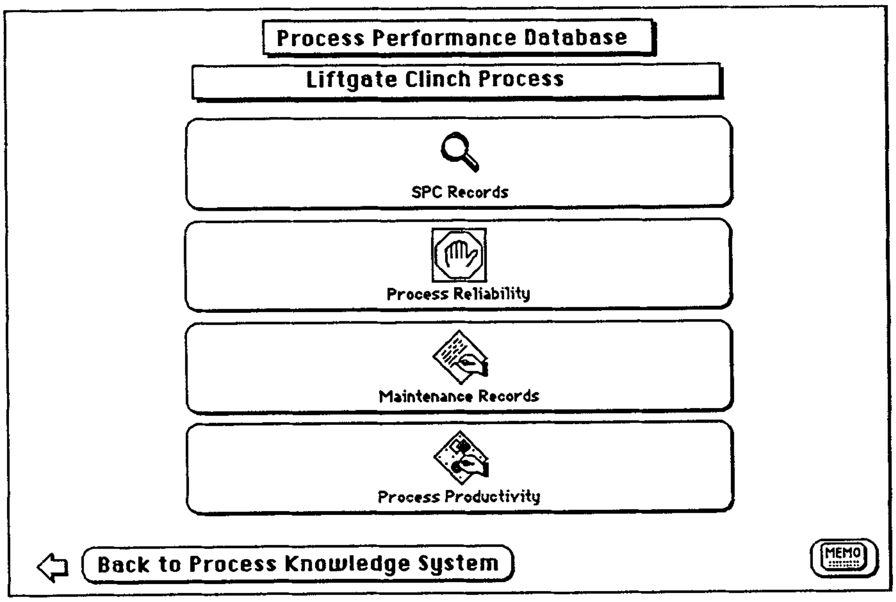


Fig. 3. Manufacturing databases.

this particular plant. Problems that involve several process steps or that occur between processes are collected under the button called "system level problem history" in the lower left of the screen. The other buttons take the user to other process databases, one of which contains performance data (e.g., throughput) and another that contains digitized forms of the engineering drawings (process sheets) for this process.

4.3.2 Next step: A second experimental system. The response to the first system was overwhelmingly positive, and as a result we immediately began work on a second, more comprehensive system, based on the initial design. We began by gathering a much more extensive and representative set of product and process performance data. We continued to work with the liftgate system, since we understood it very well by this time. We knew that the additional data we needed came from many different places, and that most of it was not computerized. The data included in the second system was entered by hand, but we plan to computerize it and bring all of it into an operational system based on

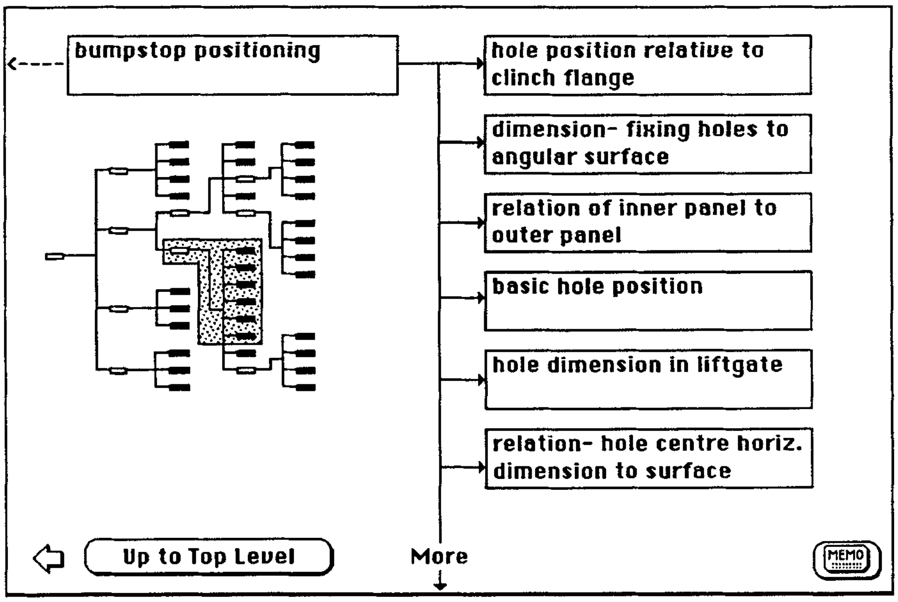
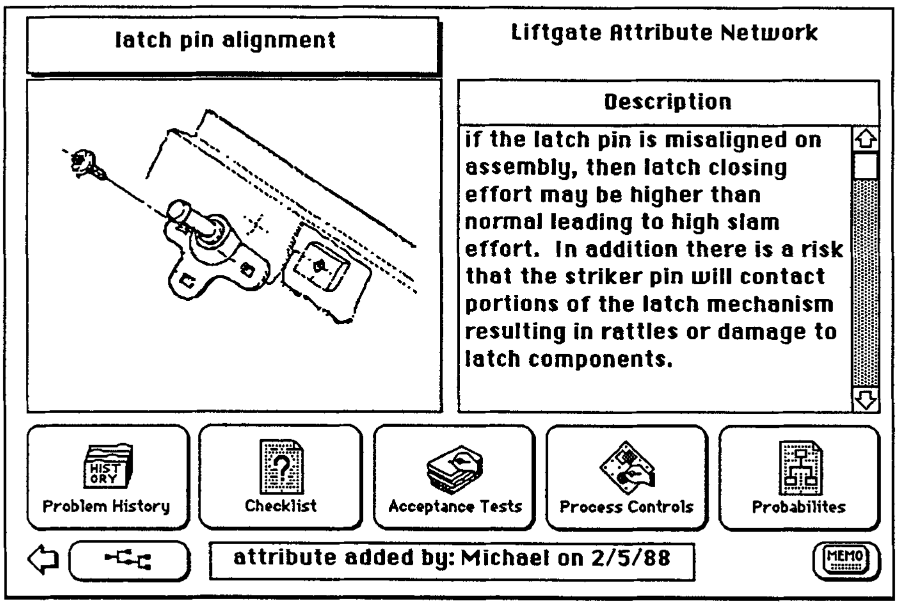


Fig. 4. Product function hierarchy.



if the latch pin is misaligned on assembly, then latch closing effort may be higher than normal leading to high slam effort. In addition there is a risk that the striker pin will contact portions of the latch mechanism resulting in rattles or damage to latch components.

attribute added by: Michael on 2/5/88

Fig. 5. Detail of critical dimension.

our second prototype. Some of the data sources, and the format they will take in the final system, are:

1. *Component drawings.* All engineering drawings of all parts of ATI's products, as well as drawings of subassemblies, will be included in graphic form in a CAD database. These drawings will be kept current and will be the authoritative source of all specifications.
2. *Assembly drawings.* Drawings by manufacturing engineers that show details of the assembly process will be in another CAD database.
3. *Factory and process layouts.* Drawings of factory floor layouts and machines, for all plants, will be kept in another database. These drawings will be cross-indexed so an engineer can look at all machines of a certain type, or all assembly lines of a certain type, or all machines at a specific factory.
4. *Product testing information.* Records kept by the product testing labs will be linked to each component or subassembly tested.
5. *Customer feedback.* A database of records from marketing surveys will contain all the complaints and compliments collected from customers. This

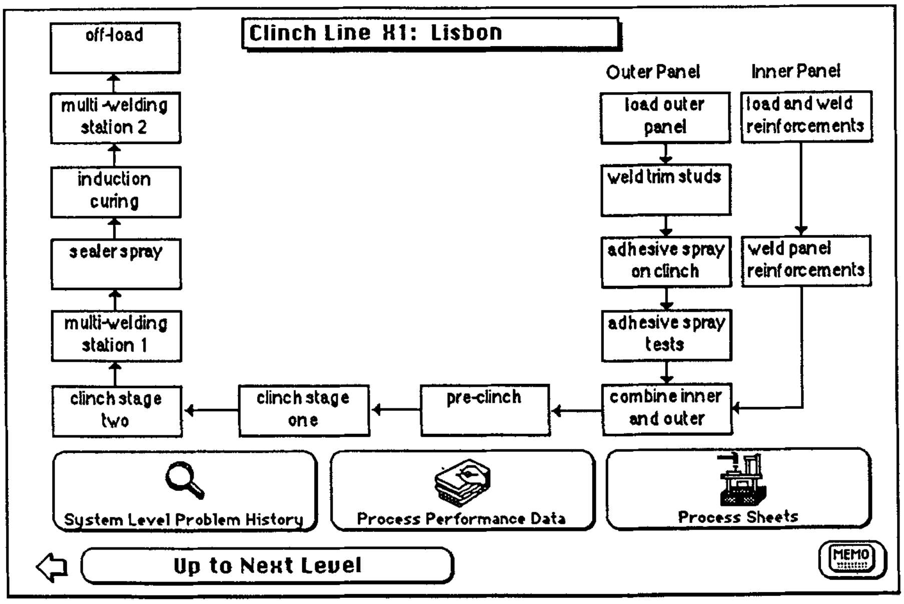


Fig. 6. Manufacturing database detail.

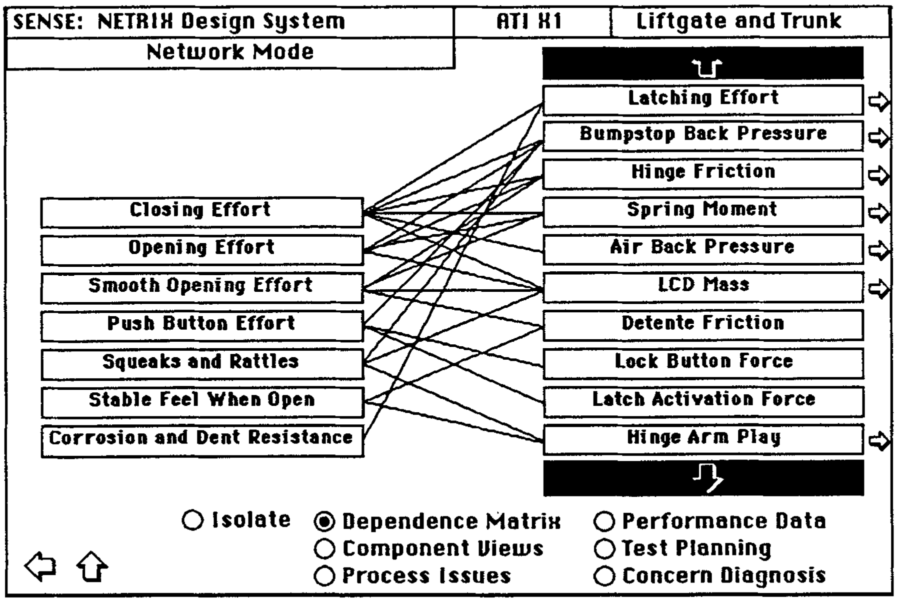


Fig. 7. Part of a liftgate functional hierarchy.

database will be cross-indexed with customer comments on competitor's products.

6. *Repair records*. The system will include records from the dealer network that describe the rates of repair and the diagnosis of each problem, organized by problem types.
7. *Plant reliability and process capability records and problem histories*. A database of records kept in each plant describes the time and duration of the problems with each plant's processes, the seriousness of those problems, and their solutions. SPC records and capability studies contain information on process capabilities.

Engineers need these types of information in order to make the best possible trade-offs in the design process.

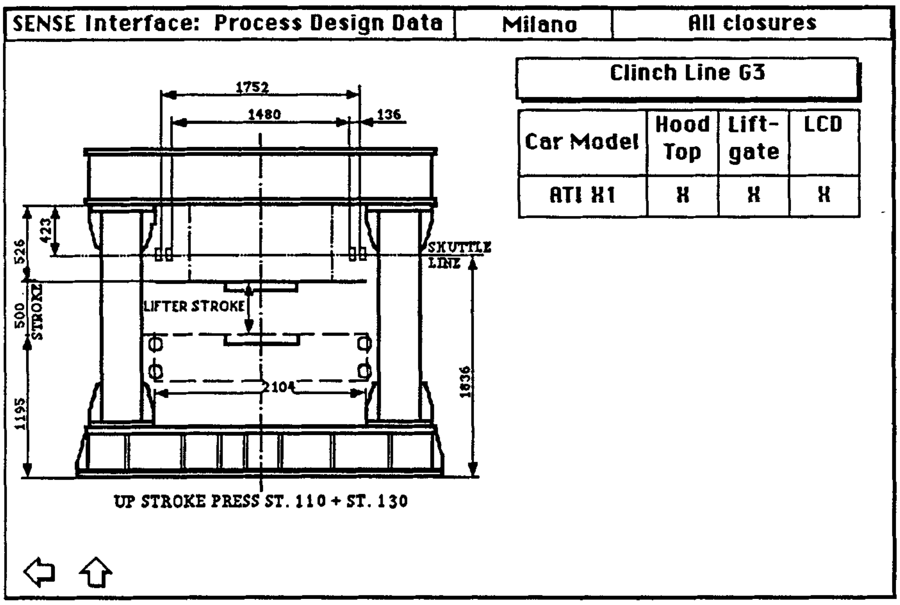
We were satisfied with our hierarchical approach to managing data access and decided to continue experimenting with it in the second prototype system. Figure 7 shows a small piece of the product function hierarchy for the liftgate as it was implemented in our second system. This hierarchy appeared in a different form, albeit with much the same content, in the first prototype (compare this figure with Fig. 4).

We attached a significant amount of manufacturing data to each engineering attribute (node) in the functional hierarchy. Figure 8 shows the type of information that can be accessed, in this case specifications for one of the presses used in a particular plant (the "Milano" plant). This press is used, as is shown in the figure, to manufacture liftgates, hoods, and trunks (LCD's) for the X1 model. Engineers in product design can use such information to determine whether a new design will be manufacturable on existing facilities, or whether new equipment will be required. As with everything else in our system, an engineer can view this drawing simply by clicking the mouse on an earlier screen. In this case, the user might have requested manufacturing information on liftgates, and then asked for clinch press data, and then chosen the Milano plant. All of these choices are made from menus, so the user is not required to memorize any commands.

Figure 9 shows another kind of process capability data included in the prototype. This data was collected by engineers in the plant, and it indicates how well various dimensions (indicated on the outline of the liftgate) conform to engineering specifications. Each of the graphs corresponds to one of the numbered points around the liftgate's perimeter (only four points are shown here; the other 12 appear on other screens, due to space constraints). These graphs show how the capability of the process changes over time.

4.4 Integrated Data Display

Implementation of the second experimental system with the richer set of data led us to consider additional interface design issues. Specifically, we found that design engineers needed "overview" displays to help them identify major design issues prior to detailed engineering work. We addressed this need by constructing screens that integrated information from several different sources.



SENSE Interface: Process Design Data | Milano | All closures

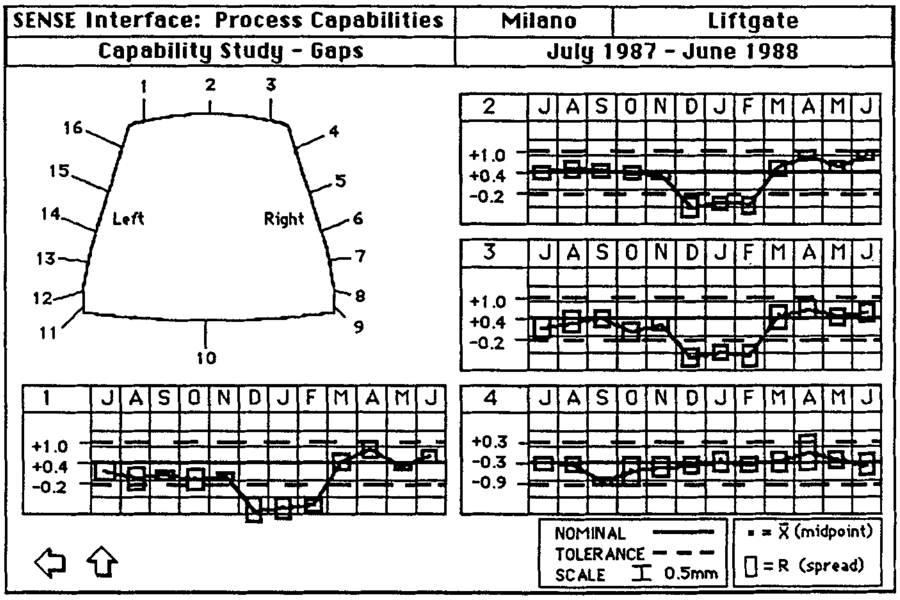
Clinch Line G3

Car Model	Hood Top	Lift-gate	LCD
ATI X1	X	X	X

Fig. 8. Clinch press drawing.

Consider the display shown in Fig. 10. It summarizes some of the product performance information about the liftgate of the ATI "X1" model, and the liftgate of a particular competitor "J1." We organized the information around the set of customer attributes (from the product function hierarchy) for the liftgate. Performance information from a number of different sources was then gathered for each customer attribute for a set of competitors' vehicles. Each of the sources of information gives engineers different insights into liftgate performance. Durability information is used to predict the performance of the liftgate as the vehicle ages. "Best in Class" studies summarize customers' reactions to competing vehicles in the show room, and the TGW and TGR studies detail the reported "things gone wrong" and "things gone right" for competing vehicles after the first 6 months of ownership. The engineer can sort this data along any of the columns simply by clicking the mouse on that column. The screen is not large enough to hold all the attributes, so the scroll bar on the right was added to allow the user to see additional ones.

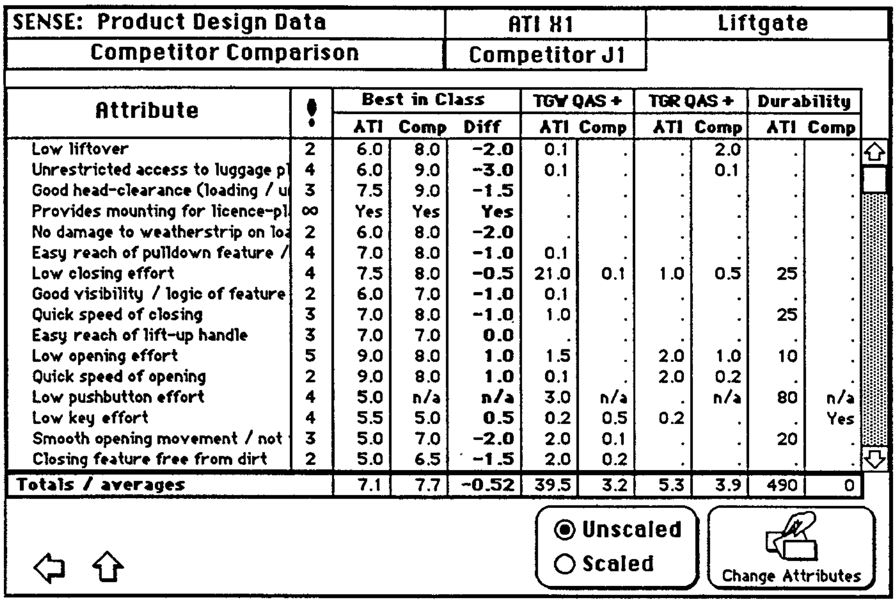
Suppose that an engineer wanted to base a new liftgate design on an existing ATI design. As a first step, he would request an overview of the perfor-



SENSE Interface: Process Capabilities | Milano | Liftgate

Capability Study - Gaps | July 1987 - June 1988

Fig. 9. Process capability, time study.



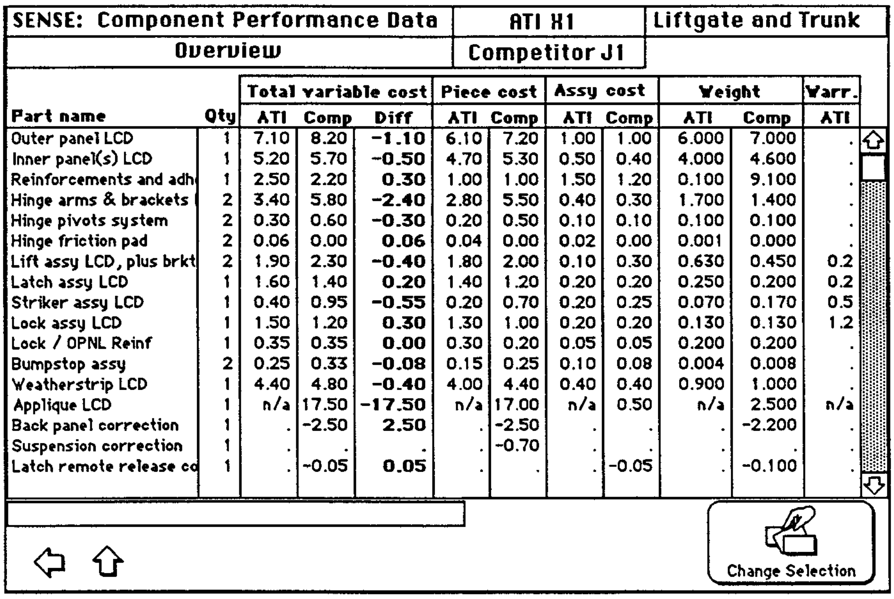
SENSE: Product Design Data	ATI H1	Liftgate
Competitor Comparison	Competitor J1	

Attribute	!	Best in Class			TGW QAS +		TGR QAS +		Durability	
		ATI	Comp	Diff	ATI	Comp	ATI	Comp	ATI	Comp
Low liftover	2	6.0	8.0	**−2.0**	0.1	.	.	2.0	.	.
Unrestricted access to luggage p	4	6.0	9.0	**−3.0**	0.1	.	.	0.1	.	.
Good head-clearance (loading / u	3	7.5	9.0	**−1.5**	.	.	.	.	.	.
Provides mounting for licence-pl	∞	Yes	Yes	**Yes**	.	.	.	.	.	.
No damage to weatherstrip on lo	2	6.0	8.0	**−2.0**	.	.	.	.	.	.
Easy reach of pulldown feature /	4	7.0	8.0	**−1.0**	0.1	.	.	.	.	.
Low closing effort	4	7.5	8.0	**−0.5**	21.0	0.1	1.0	0.5	25	.
Good visibility / logic of feature	2	6.0	7.0	**−1.0**	0.1	.	.	.	.	.
Quick speed of closing	3	7.0	8.0	**−1.0**	1.0	.	.	.	25	.
Easy reach of lift-up handle	3	7.0	7.0	**0.0**	.	.	.	.	.	.
Low opening effort	5	9.0	8.0	**1.0**	1.5	.	2.0	1.0	10	.
Quick speed of opening	2	9.0	8.0	**1.0**	0.1	.	2.0	0.2	.	.
Low pushbutton effort	4	5.0	n/a	**n/a**	3.0	n/a	.	n/a	80	n/a
Low key effort	4	5.5	5.0	**0.5**	0.2	0.5	0.2	.	.	Yes
Smooth opening movement / not	3	5.0	7.0	**−2.0**	2.0	0.1	.	.	20	.
Closing feature free from dirt	2	5.0	6.5	**−1.5**	2.0	0.2	.	.	.	.
Totals / averages		7.1	7.7	−0.52	39.5	3.2	5.3	3.9	490	0

Fig. 10. Overview of liftgate functional attribute data.

mance of the liftgate in comparison to competitors. The integrated presentation of several performance measures, as shown in Fig. 10, allows the engineer to recognize patterns than might go unnoticed if single performance measures were presented. Consider, for example, the data for the "low closing effort" attribute. Closing effort is the effort required to close the tailgate. Although "show room" performance and durability tests of closing effort were satisfactory, relatively large number of customers complained about closing effort after 6 months of ownership (see the TGW column). Since the engineer would know that show room and durability testing both use small samples tested with nominal conditions, this data might suggest to him that the problem is due to excessive variability in the manufacturing process. He could then investigate in more detail the implications of this problem for the design of a new liftgate, possibly using the database of manufacturing information.

Figure 11 shows a screen that provides the engineer with an overview of cost and warranty data for all the components of the liftgate and the trunk (these two subsystems share many components). Displays of this type provide ATI's engineers with a tool that helps them to understand how their cost



SENSE: Component Performance Data	ATI H1	Liftgate and Trunk
Overview	Competitor J1	

Part name	Qty	Total variable cost			Piece cost		Assy cost		Weight		Warr.
		ATI	Comp	Diff	ATI	Comp	ATI	Comp	ATI	Comp	ATI
Outer panel LCD	1	7.10	8.20	**−1.10**	6.10	7.20	1.00	1.00	6.000	7.000	.
Inner panel(s) LCD	1	5.20	5.70	**−0.50**	4.70	5.30	0.50	0.40	4.000	4.600	.
Reinforcements and adh	1	2.50	2.20	**0.30**	1.00	1.00	1.50	1.20	0.100	9.100	.
Hinge arms & brackets	2	3.40	5.80	**−2.40**	2.80	5.50	0.40	0.30	1.700	1.400	.
Hinge pivots system	2	0.30	0.60	**−0.30**	0.20	0.50	0.10	0.10	0.100	0.100	.
Hinge friction pad	2	0.06	0.00	**0.06**	0.04	0.00	0.02	0.00	0.001	0.000	.
Lift assy LCD, plus brkt	2	1.90	2.30	**−0.40**	1.80	2.00	0.10	0.30	0.630	0.450	0.2
Latch assy LCD	1	1.60	1.40	**0.20**	1.40	1.20	0.20	0.20	0.250	0.200	0.2
Striker assy LCD	1	0.40	0.95	**−0.55**	0.20	0.70	0.20	0.25	0.070	0.170	0.5
Lock assy LCD	1	1.50	1.20	**0.30**	1.30	1.00	0.20	0.20	0.130	0.130	1.2
Lock / OPNL Reinf	1	0.35	0.35	**0.00**	0.30	0.20	0.05	0.05	0.200	0.200	.
Bumpstop assy	2	0.25	0.33	**−0.08**	0.15	0.25	0.10	0.08	0.004	0.008	.
Weatherstrip LCD	1	4.40	4.80	**−0.40**	4.00	4.40	0.40	0.40	0.900	1.000	.
Applique LCD	1	n/a	17.50	**−17.50**	n/a	17.00	n/a	0.50	n/a	2.500	n/a
Back panel correction	1	.	−2.50	**2.50**	.	−2.50	.	.	.	−2.200	.
Suspension correction	1	.	.	.	.	−0.70	.	.	.	.	.
Latch remote release co	1	.	−0.05	**0.05**	.	.	.	−0.05	.	−0.100	.

Fig. 11. Overview of liftgate structural data.

and quality priorities compare with those of competitors. In the display shown below we see that ATI spends much less than competitor J1 (the market leader) on hinge arms and brackets. This might point to a competitive advantage in manufacturing at ATI. However, in combination with the information on closing effort problems discussed above, it may point to a willingness on the part of the market leader to spend more on a critical component to obtain superior performance. Product and process engineers want to obtain this type of overview of the current design situation, and clear presentation of such overviews is essential to their acceptance of a system[6].

5 Barriers to Implementation

Although our efforts to date have proven quite successful, we would like to outline some of the organizational and technical barriers that we have encountered thus far. Creation of a uniform engineering database in a large company is a formidable challenge. We will not know for several years if we will be successful, since the system we are describing here represents a radical change from the way ATI currently does its engineering.

5.1 Organizational Barriers

In building our experimental system, we encountered a number of organizational barriers to the implementation of engineering databases. In this section we describe the barriers that we encountered, and our recommendations for addressing them[7].

5.1.1 Motivating the engineer to contribute to the system. The databases that we have described thus far appear to the engineers as a single "virtual" database, to which all of them have (at least) read access. However, many of ATI's engineers will have to take responsibility for entering and maintaining data. In order to motivate the engineering groups to maintain their portion of the database, someone must convince them that the resulting system will be beneficial to them personally. This is a key point: if a system developer tells an engineer that the database he maintains will be useful to another engineer (someone perhaps far removed from him), he will not necessarily be motivated to keep his data current. The company must either convince the engineers that it is to their advantage to maintain the data, or else hire full-time people to maintain it. The former choice is clearly preferable.

5.1.2 Achieving a critical mass of data and users. User acceptance of an engineering database system appears to be subject to a threshold effect, of the following nature: if the amount and quality of data in the system exceeds some threshold, then the system is of tremendous benefit to users. It becomes their primary information source. Below the threshold, users must still gather a significant amount of information from other sources. As a result, they will use the database less frequently. If the interface is poorly designed, they will have to reacquaint themselves with the system each time they want to use it. If a critical mass of users are not willing to contribute and maintain their own data, then the value of the database will be insufficient to justify its construction. We are addressing this problem by focusing efforts in data collection, model building, and training on one vehicle system at a time. Groups of users that are tightly coupled in their use of information are therefore brought up to speed together. In this way, we hope to attain "local" critical masses of data and users. Later, more experienced groups of users can assist in the training of new users.

5.1.3 Existing reward systems. Existing reward systems in a company may act as barriers to the implementation of performance-assessment systems. Oddly enough, the more accurate and comprehensive the engineering database is, the more the existing reward systems can be a problem. Consider, for example, the reward system for plant managers at ATI. Industrial engineers assign "standard" production capacities to all new processes within a plant. These capacity estimates are updated on a periodic basis, but since accurate process performance information is not available, these estimates may be inaccurate. The performance of the plant relative to the standards is the primary measure by which upper management evaluates plant performance. As a consequence, plant managers have strong incentives to push for conservative initial estimates and then to "manage" a steady increase in plant performance over a period of years. An accurate, comprehensive process performance database, however, would essentially "open up" the actual performance of individual processes

---

[6] For a discussion of the importance of "situation assessment" in the design of decision support systems, see Rouse [1988].

[7] Researchers in organizational behavior and change management have recommended specific approaches for the more general problem of introducing new computer systems in a firm. See Leonard-Barton [1988].

in the plant to scrutiny by upper management and industrial engineers. Since industrial engineers could then make much more accurate estimates of the capacities of particular processes, it might be more difficult for plant managers to show steady improvements.

5.1.4 Information as power in the organization. Paradoxically, open access to information is most valuable to people at the top and bottom of the ATI organization. Top management gains because they are most concerned about external competitive performance, which should improve with better communication and performance measurement. Design engineers gain by getting better tools for doing concrete design work, and becoming less subject to the influence of middle management on design issues.

Unfortunately, there seems to be less benefit in open access for middle management. Despite the fact that a database of engineering information with open access would vastly improve the speed and quality of communications, it offers fewer benefits, and may even threaten, the very managers whose cooperation is needed to create it. This results from the fact that much of the hierarchy in any large company is devoted to filtering and passing along information. (In fact, corporate hierarchies were created many years ago in part to manage the massive amount of information that needed to be communicated within a large company. Computer technology, however, provides a much better information management tool.) At ATI, if an engineer has a problem or a request which needs to be communicated to an engineer in another area, he goes through channels of management: he tells his boss, who may handle it himself or pass it up another layer. With our engineering database, many of these requests for information will be unnecessary. In addition, records of problems in manufacturing or in the field will be entered almost immediately (e.g., daily) into the database, and thereby communicated to all the affected engineers. The interface will include an electronic mail system which will allow an engineer to notify any other engineer directly for more urgent manners. All of this curcumvents normal organizational channels.

As in many large manufacturing companies, the exclusive access to information within ATI carries with it considerable power. For those without other sources of power, open access will be viewed as a threat. We observed the pathological implications of this phenomenon most clearly at the interface between the product design and manufacturing functions. Each function wants free access to information possessed by the other, but would like to control access to its own information. An excellent example arises in the handling of problems in ATI's manufacturing plants. In the present system, problems often result in much finger pointing, with product design engineers claiming that manufacturing failed to build the product correctly, and manufacturing claiming that the product was poorly designed. In such an environment, an open database can be used as easily to find evidence of someone else's culpability as to search for solutions. Senior managers in both product design and manufacturing fear open access for this reason. Also, an open system really does make it harder to avoid having mistakes (even inconsequential ones) become public knowledge. Once again this can be painful if abused.

To be fair, we should note that ATI's managers have had negative experiences with systems technology, in which technology was promised and not delivered, or delivered in a very different form from what they expected. Hence a "show-me" attitude is, to some degree, understandable. Database developers must be prepared to invest in building credibility with all levels of management, and to manage carefully the expectations of all user groups.

5.1.5 Information systems infrastructure issues. Many existing databases may appear to address engineering information needs. Unfortunately, most such databases were developed for managerial control purposes (usually cost control), and not for engineering or quality assessment purposes. Therefore, much of the existing data is often not of the appropriate type of "granularity" for design engineers. Consider the existing warranty database: ATI has an enormous database containing information on the repair frequencies for every component of every ATI vehicle. The data is gathered from dealers and aggregated into "repairs per 100" numbers. Beyond its existing role of highlighting problem components, it is of little value to design engineers. Engineers need more details on the reasons why components fail, details that are not gathered in a systematic fashion at present.

However, there are strong cost pressures to make use of existing databases. The objections of existing database "owners" are reasonable in the context of the environment in which hardware and software decisions are made at ATI. The purchase of a new computer system almost always benefits the person responsible: approval of the system represents approval of his plan for it, which usually means he gets recognition as an innovator. He may also get additional resources, in the form of budgetary control and personnel, which raises his status in the organization. It is only natural, therefore, for the owners of existing databases to view with suspicion the motives of anyone wanting to replace their computer systems. In this fashion, the existing hardware and software itself creates barriers to innovation.

5.1.6 Proprietary information issues. ATI, just like virtually all automobile manufacturers, uses a large number of externally supplied items in its vehicles. Electronic systems for engine control, for example, are purchased from outside suppliers. The overall performance of ATI's products is dependent on interactions between externally supplied and internally produced items. Thus, to control product performance accurately, ATI must have access to product and process information from these supplier firms. This information may be of proprietary value to the supplier, who may therefore be reluctant to supply it. On the other hand, in order to design products that meet ATI's requirements, the supplier should have access to at least some internal manufacturing information. The issue of whether or not suppliers should have access to the engineering database presents both technical and competitive challenges.

5.2 Technical Barriers

Even without the organization barriers just described, many technical barriers stand in the way of implementing an open engineering database in a large company like ATI.

5.2.1 Heterogeneity. The largest technical barrier is the integration of data from dozens of different databases, each using incompatible software and hardware, and typically involving different database semantics. Although entering data that is not yet computerized requires considerable effort, it is much easier to find resources for data entry than it is to convince the owners of an existing database to convert their data *and* data collection systems for the benefit of a group of engineering users. Despite much talk of heterogeneous databases in the computer science research community, very few commercial solutions exist. Typically, special software must be developed to allow any two databases to communicate with one another.

5.2.2 Flexibility through changes. The requirement for flexibility is a key technical barrier in the development of engineering database systems. Automotive companies like ATI periodically introduce completely new car models. Prior to the launch of a new model, manufacturing facilities may undergo extensive alterations. To be useful, engineering information must remain "live" in the sense that it accurately reflects the current situation. An engineering database must therefore be capable of adapting to many different large and small changes in ATI's products and processes. System developers and users must be able to create and alter domain models, and reorganize the sets of queries that support access to the database. In addition, the system must support the creation of linkages between the old state of the process and the new state. An engineer should, for example, be able to view the history of an individual machine, even though it may produce many different components over its lifetime, and may have been associated with several different manufacturing processes, or even different plants.

5.2.3 Data classification and aggregation. Problems of data classification and aggregation loom large in preventing the effective use of product and process performance data. Consider, for example, the market research data for the liftgate that we discussed above. A large volume of this data consists of verbatim transcripts of interviews with customers. A certain amount of classification and aggregation is necessary to reduce the raw data to a usable format. This data must then be integrated with test data from a variety of sources. The test data also must be classified. Because this classification process is done manually, and those classifying the data may not have the expertise to those using it, a certain amount of misclassification is inevitable. The users are able to recognize the misclassifications, but rarely view the raw data. If misclassification is recognized, there should be a mechanism for reclassifying individual data points and obtaining "what if" changes in aggregate results.

5.2.4 Communications and data storage requirements. The communication and data storage requirements of our system represent another technical barrier. Data will be entered at a number of geographically dispersed sites. This data must be rapidly accessible from all other sites[8]. This raises issues of how the data will be stored and accessed. Should the data be stored in a decentralized manner or in a large central unit? Because of the long distances involved, and the amounts of data, dedicated

---

[8] Since we are not dealing with control data, there is no need for real-time updating and access. Only a limited subset of the data needs to be maintained daily.

communications links may be required. Since we expect every engineer to be using the system every day, we need to be able to assure ATI that the system will be functional a very high percentage of the time. We need a system that is insensitive to problems with communication lines and local area networks, one which allows the engineering organization to be productive even if access to remote sites is lost.

6 New Database Technology Requirements

At present we are investigating the potential of new kinds of database systems to address our needs for richer data models, increased system flexibility, and integration of multiple data sources (see Rychener et al. [1986] for one approach to integrating disparate data types). The most important capabilities for our purposes include:

- strong support for distributed access, concurrency control, and crash recovery, equivalent in reliability and performance to the best of the present generation of relational database systems
- transparent read-only access to existing remote databases, including the integration of multiple data sources in response to a query, and the ability to rapidly modify such queries
- a version control mechanism that would support the preservation of (for example) data for different configurations of a manufacturing process as it evolves over time
- domain modeling capabilities that allow developers to embed significant semantic content in the domain model, rather than in the application programs that use the database
- features that make the database more active; for example, a mechanism for alerting an engineer automatically when a specific datum changes

At present, we plan to use a rapid prototyping environment, in conjunction with an object-oriented database, to create the functional and structural hierarchies for query management, to generate and execute queries, to create the user interface and integrated displays, and to support user navigation through the database.

Other researchers in this area have designed systems that, although quite different from ours, offer many of the same benefits [e.g., Kilhoffer and Kempf 1986, Mittal and Araya 1986, Cutkosky and Tenenbaum 1989, Friel et al. 1989]. The growing body of research on the problem of systems for engineering design provides a convincing argument in itself that database technology must evolve to meet some of the needs of large manufacturing companies.

7 Conclusion

The purpose of this paper has been to describe the progress of a research project in which something completely new to the manufacturing world is being created. A heterogeneous database with a single interface, openly accessible to any of several thousand engineers at widely scattered sites, has never been created on a scale such as we are proposing. It represents a radically different way of managing engineering information, one which cuts across all the traditional lines of management and consequently faces many barriers. Despite these barriers, this database architecture, and the style of working that accompanies it, offers the possibility of significantly improved engineering quality, reduced cost, and the avoidance of countless problems that occur today.

We present this work in the hope that we may cast light not only on some important issues in engineering management, but on database design itself. We have not yet found a commercial database product that offers features to aid us in our task of integrating dozens of different databases. This lack of compatibility may be a natural consequence of the fact that databases are created by companies in a competitive market, but there is no reason why it cannot be overcome. Developments in object-oriented database technology show great promise for addressing the challenges of developing engineering information systems. We hope the problems presented here will stimulate thought on new database architectures. It has become clear to us in our research that manufacturing companies such as ATI are moving inexorably toward a computerized engineering environment, and we hope to offer them something more integrated and easier to use than the hodgepodge of tools available today.

Acknowledgments. The authors wish to express their appreciation for the support and guidance of Professor Ramchandran Jaikumar of the Harvard Business School.

References

Clark, K. B. The interaction of design hierarchies and market concepts in technological evolution. *Research Policy,* 14, 1985, 235–251

Clark, K. B., W. B. Chew, and T. Fujimoto. Product development in the world auto industry. *Brookings Papers on Economic Activity,* 3, 1897, 729–771

Clark, K. B. and T. Fujimoto. Overlapping problem solving in product development. Harvard Business School Working Paper 87-049, 1987

Clark, K. B. and T. Fujimoto. Lead time in automobile product development: explaining the Japanese advantage. Harvard Business School Working Paper 89-033, 1989

Cutkosky, M. and J. Tenenbaum. A methodology and computational framework for concurrent product and process design. *Mechanism and Machine Theory,* Vol. 24 1989

Dussalt, H. B. The evolution and practical application of failure modes and effects analysis. RADC-TR-33-72, In-House Report of the Rome Air Development Center, March 1983

Friel, P., R. Mayer, J. Lockledge, G. Smith, and R. Schulze. Coolsys: a cooling systems design assistant. In H. Schorr and A. Rappaport (eds.), *Innovative Applications of Artificial Intelligence,* Menlo Park: AAAI Press, 1989, 203–212

Hauser, J. and D. Clausing. The house of quality. *Harvard Business Review,* May–June 1988, 63–73

Hutchins, E., J. Hollan, and D. Norman. Direct manipulation interfaces. In D. Norman and S. Draper (eds.), *User Centered System Design,* Hillsdale: Erlbaum Associates, 1986, 87–124

Jaikumar, R. and R. Bohn. The development of intelligent systems for industrial use: a conceptual framework. Harvard Business School working paper 9-786-024, 1986

Kilhoffer, A. and K. Kempf. Designing for manufacturability in riveted joints. *Proceedings of AAAI-86,* Los Altos: Morgan Kaufmann Publishers, 1986, 820–824

Leonard-Barton, D. Implementation characteristics of organizational innovations: limits and opportunities for management strategies. *Communications Research* 15:5, 1988, 603–631

Mittal, S. and A. Araya. A knowledge-based framework for design. *Proceedings of AAAI-86,* Los Altos: Morgan Kaufmann Publishers, 1986, 856–865

Nau, D. S. Hierarchical abstraction for process planning. *Second International Conference on Applications of Artificial Intelligence in Engineering,* 1987

Rouse, W. B. Intelligent decision support for advance manufacturing systems. *Proceedings of Manufacturing International,* Vol. II, ASME, 1988

Rychener, M., I. Hulthage, M. Fox, and M. Farinacci. Integration of multiple knowledge sources in ALADIN, an alloy design system. *Proceedings of AAAI-86,* Los Altos: Morgan Kaufmann Publishers, 1986, 878–882