Effective Prototyping during Product Development

by

Hans Patrick Griesser
S.B. Mechanical Engineering, MIT (1983)

Submitted to the Departments of
Mechanical Engineering and Management
in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Mechanical Engineering
and
Master of Science in Management

in conjunction with the
Leaders for Manufacturing Program
at the
Massachusetts Institute of Technology
June, 1992

Signature of Author Signature redacted
Department of Mechanical Engineering
Department of Management Science
May 8, 1992

Certified by Signature redacted
Warren P. Seering
Professor of Mechanical Engineering

Certified by Signature redacted
Karl T. Ulrich
Assistant Professor of Management

Accepted by Signature redacted
Ain Sonin
Chairman, Department Committee

Effective Prototyping during Product Development

by

Hans Patrick Griesser

submitted to the departments of
Mechanical Engineering and Management
on May 8, 1992
in partial fulfillment of the requirements for the degrees of
Master of Science in Mechanical Engineering
and
Master of Science in Management

ABSTRACT

Using prototypes effectively can be a difficult task. There is little agreement on the best sequence of prototype steps to take or the prototyping processes to use for a given part's design and development. This work uses observations of plastic part development and prototype use, which were made during an internship at the Kodak Apparatus Division, as the basis for a model of prototype use during the product development process. The observations and the model are used to generate metaphors and heuristics to help designers improve their product development processes. The model may also be used as a communication tool to focus a product development team's effort.

The model views product development as a process of reduction of concern levels of the product's attributes. Prototypes, broadly defined, are the tools used to reduce the concern attributes. A computer code version of the model was written and used to search for the sequence of prototype processes which most effectively reduced a given set of starting concern levels. The code can easily be modified to evaluate a particular sequence or evaluate heuristics.

The insights from the study can be summarized in these statements:

- Learn fast and keep moving
- Reduce product risk
- Plan ahead

Future needs for prototyping processes identified were those that can more quickly produce richer part representations. Strategies to streamline the prototype production process are also valuable.

Thesis Supervisors:

Warren P. Seering, *Professor of Mechanical Engineering*
Karl T. Ulrich, *Assistant Professor of Management*

Acknowledgments

Well over a year went into researching and writing this thesis. There are some important people who helped me along the way that I would like to thank.

First, Karl Ulrich and Warren Seering as advisors and teachers were the best. Together they taught me to think more critically and clearly. I am grateful for their time, attention, suggestions, and patience.

The Eastman Kodak Company was very gracious to let me use their facilities and talk to their people. I especially want to thank my advisors, Ron Caldwell and Glenn Tersteeg, for their efforts and suggestions. I am also grateful to all the engineers, designers, and prototype makers that took time to meet with me and share their problems and successes.

Andy Christian was an enormous help with the computer coding. Without his help, my code would have been twice as long and much harder to use and understand.

Anne Sinclair is a great friend, encourager, listener, helper, and editor. She inspired me to keep working, finish the thesis, and do the best I can. I would also like to thank her family for the *Star Trek* reference.

My family has been ever supportive of me. I love them all very much.

Finally, the Leaders for Manufacturing program paid for all this and more. The program was flexible enough to let me start my internship early so I could be on campus later to take some special classes. Here is their official thank you:

> *The author gratefully acknowledges the support and resources made available to him through the Leaders for Manufacturing Program, a partnership between MIT and major U.S. manufacturing companies.*

Contents

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION: A PROJECT TO EVALUATE AND MODEL PROTOTYPE PRACTICE

One of the most important tools used in product development is the prototype. A prototype is a representation of the product that approximates the final production version of that product. Prototypes are used to test ideas such as: will it work and will it satisfy the customer. They are also used to communicate product attributes to the product team, management, and the customers.

Prototypes can be divided into two groups: physical and analytical. Physical prototypes include: mock-ups, breadboards, and Alpha/Beta test units. Analytical prototypes include: equations, simulations, and computer models. One can also consider part drawings or CAD representations as prototypes; these fall somewhere between the physical and the analytical types.

There are many processes available to make prototypes. In fact, there is a relatively new industry called "rapid prototyping" that constructs physical parts automatically from the data of a 3-D CAD model [Brown91, Deitz90, Leonard91, Machlis91, Miller91]. In the rapid prototyping venue alone, there are several different technologies available to make parts. With all these choices available, it is important for the designer to be aware of the advantages of the different technologies.

This thesis examines the problem of effectively using prototypes throughout the product development processes. How does one think about prototyping before product development gets underway? One way to visualize the strategy choices for a product development manager is the node map in Figure 1. Each node represents a prototype, a testing point for the product. The arcs represent the different processes that can be used to get to the node. The product development manager must decide which processes (arcs) to use and the uses for the prototypes (which nodes to go through). Factors that can affect the choices include the available processes, past practices, familiarity with a process, project risk, and others discussed below. This thesis addresses the problem of finding the best way through the entire network.

This work is the result of a seven month internship I did on-site at the Kodak Apparatus Division in Rochester, NY as part of the Leaders for Manufacturing program.

A PROTOTYPE REPRESENTS A PRODUCTION PART

If a picture is worth a thousand words, a prototype is worth a thousand pictures. A prototype is defined as a representation of the part that approximates some of the attributes of the production version. Usually, a prototype is considered something physical, like a machined part. Actually my definition of a prototype is broader than just a physical entity; it includes representations ranging from mental pictures or sketches of the product, electronic (2-D and 3-D CAD) files, physical "mock-ups", and even the parts made with the production equipment to see where it must be "tweaked" to be right. This work focuses primarily on plastic molded parts for which the following types of prototyping processes were used:

- Electronic representation
  (CAD, FEA, etc.)
- Physical representations using alternative processes or material
  (CNC, Stereolithography, etc.)
- Molding type processes
  (Rubber mold, Production mold, etc.).

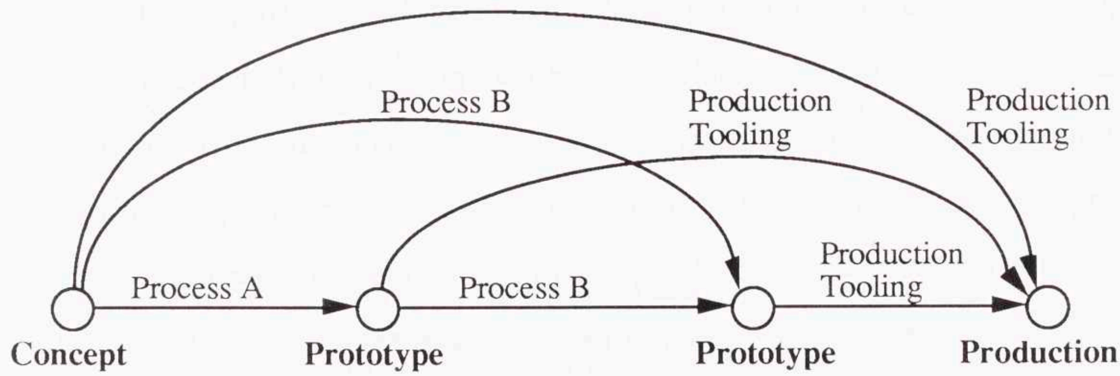


Figure 1. Prototyping Process Choices during Product Development
The nodes are prototypes. The arcs are processes to make the prototypes. What is the best path to get from Concept to Production?

MOTIVATION FOR THE PROJECT

Planning has always been important in product development. But good planning is critical to reduce product development time and be competitive. Many firms are formalizing their product development process. The rationale is that if the product developers have a plan to follow, they can reduce the overall development time and increase product quality [Clausing91]. Decisions about prototyping are important ones in the product development process.

To examine the way engineers, designers, and managers think about a sequence of prototypes, I ran some small, simple tests. During presentations of my work, I asked my audience what types of prototypes they would use for the part shown in Figure 2. Each member was given a marker and a transparency sheet with a grid on it. The vertical axis of the grid contained prototyping processes; on the horizontal axis were numbers representing the order of prototypes. I asked each person to draw lines between the points on the grid to show the order of prototyping processes they would use for the part. When the audience was finished, I collected the transparencies, aligned them in one pile, and displayed it on the screen. A typical output is shown in Figure 3. Many different paths were chosen. I got similar results after testing several different groups. Although this is not a rigorous test, it is strong anecdotal evidence that different people think differently about prototyping and the processes used.

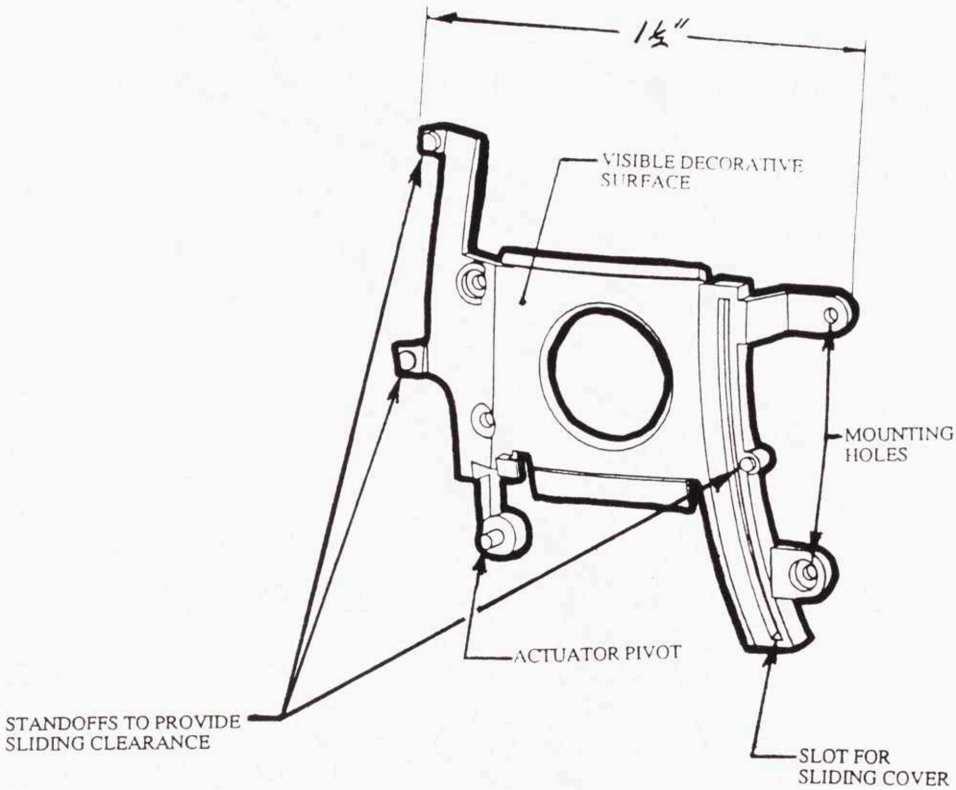


Figure 2. **Part for Prototype Sequence Test**

Participants were asked the sequence of prototyping processes they would use between concept and production of this injection molded part given the following information:

Function: Camera Part. Holds the lens cover system in place. Middle circle is a decorative part because it is visible when the lens cover is open. Long slot on right is for the sliding cover to move in. Large post at left is pivot point for cover actuator. Three small posts are standoffs to give cover clearance to move. Four screws hold the part in place.

Critical Characteristics:

- Height of standoff
- Width of slot
- Location of pivot post

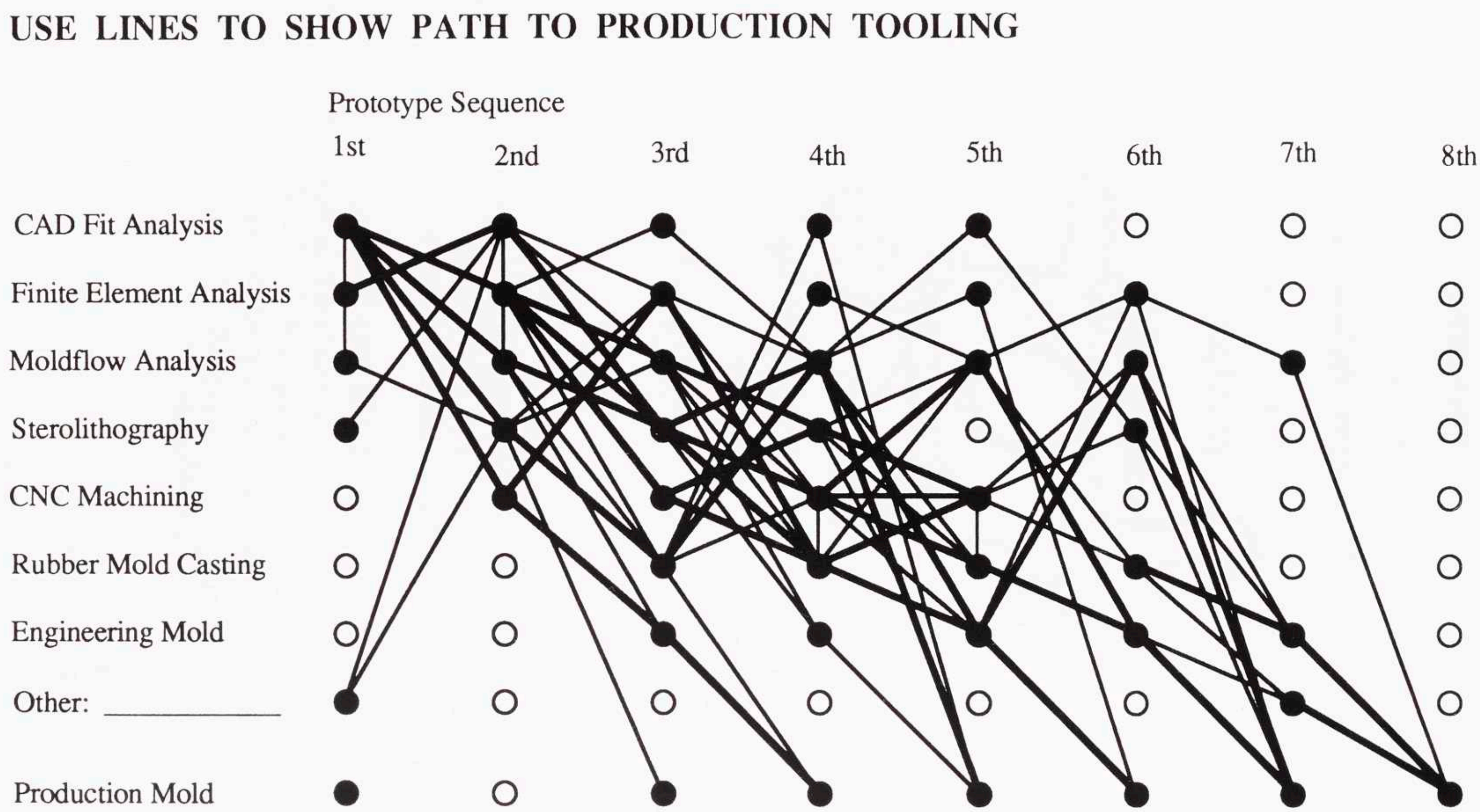


Figure 3. Typical Test Output:
The Many Different Paths Suggested to Prototype the Part
The results of 33 responses. The thicker lines mean that more than two responses included that path segment.

Another indicator of the difficulty of the problem was a survey quiz that I sent to approximately 35 engineers, designers, and model-makers at Kodak. The quiz consisted of questions about prototyping five different parts. Even though every person I asked agreed to complete the survey, I only received three responses, and two of these were incomplete. The non-participants complained that the test was too open-ended They could not easily identify the sequence of prototypes to use because of all the possible complications. Yet the questions were no more open-ended than actual product development: "Here is a part. What problems might you encounter? How are you going to test the design before committing to production tooling?" The point is that because planning a development sequence is difficult, it is often not done.

PROJECT GOALS AND DESIRED INSIGHTS

The overall goal of this project was to understand how prototypes can be used effectively during the product development process. An important part of that goal includes developing a model that describes prototype use in product development. This model can be the basis for evaluating the strategies observed in practice and other alternative strategies.

The goals of the project and the model are listed in Table 1 and described below.

Reduce Total Product Development Risk and Cycle Time

The primary goal of this project is to gain insight and develop a better understanding of how parts are prototyped. The hope is that this understanding will help a designer make the optimal decision that balances risk and development time. The prototyping decision should let one optimize the total development time and risk instead of sub-optimizing a particular step in the development process.

Determine the Best Sequence for a Particular Part

Once there is insight into a problem, one can begin to optimize the situation. The optimal solution this model hopes to find is the best sequence of prototyping activities that reduces the risk of making the wrong part and reduces the total product development time. Over the course of my internship, this optimization problem was stated several ways by my academic advisors, Kodak advisors, and me:

- "Find a concise method for defining a prototype path that leads to production processes."
- "What is the best sequence of prototyping activities to develop a part from concept through production? For example, is it better to do lots of CAD work or go right to making the production mold?"
- "Make a decision tree for the best prototyping sequence to go from part concept to production injection molds."
- "How can we learn all we need to know about a design with the fewest prototypes (or least time or least cost)?"
- "Determine the most effective sequence of prototyping processes to use for a given part."

Table 1. Project and Model Goals

- Reduce Total Product Development Risk and Cycle Time
- Determine the Best Sequence for a Particular Part
- Determine the Effect of a Prototype Process in a Sequence
- Extract Heuristics from Observations and the Model's Behavior
- Understand the Value of a Prototype Process
- Add to the Understanding of Manufacturing Systems

Determine the Effect of a Prototype Process in a Sequence

I want to understand the effect a particular prototype has on the product development cycle. The part being developed and the prototyping process must be characterized in a way that makes sense to the designer. The interaction of the part characteristics and the prototype process's abilities will determine the usefulness of that prototype. By modeling these, I hope to gain insight into the sequence of prototyping activities used to develop a part from concept through production.

Extract Heuristics from Observations and the Model's Behavior

An important benefit from my observations of prototype practice, the model insights, and the method to determine the optimal sequence strategy is a list of guidelines or heuristics. These would help a designer or manager look at a particular part and quickly determine a sequence of prototyping activities to develop it. The heuristics would not necessarily always give the best solution, but they should give a very good solution in most cases.

Understand the Value of a Prototype Process

The prototyping field has grown rapidly over the last decade because of the proliferation of more powerful CAD packages, automatic computer analysis programs, and the ability to use CAD files directly for CNC machinery, stereolithography, and rapid prototyping technologies. There is a need to understand the value of these new abilities to the product development cycle. From developing and exercising the model, I hope to understand prototyping processes better. For example, the model could help do the following:

- characterize existing and potential prototyping processes for their ability to address certain part attributes, the learning rate when using the prototype, and the nominal time to make that type of prototype.
- recommend prototyping processes with particular abilities to improve the product development cycle.

Add to the Understanding of Manufacturing Systems

Finally, this is a Leaders for Manufacturing thesis, and it is important to note that the prototype sequencing model will help the continuing effort to characterize the "laws" of manufacturing. This is an attempt to understand the complex interaction of people and processes in a product development group that is part of a larger manufacturing organization. The ultimate goal is to capture the essence of the prototyping process and perhaps be part of the "creation of new fundamental knowledge" [Little92].

METHODOLOGY OF THE PROJECT

The research for this thesis was done during a seven month on-site internship at Kodak at their Apparatus Division (KAD). KAD is where most of the mechanical parts for Kodak's products are made.

During the internship, I interviewed engineers and designers about their prototyping philosophy. Detailed part histories were collected and analyzed. These are compiled in Appendix A.

To learn about prototyping processes, I interviewed model makers and machinists. They provided the process practitioner's view of prototyping. I also tried several of the prototyping processes myself to get a feel for the technologies.

The observations of practice and from the literature were synthesized into a model that describes how prototypes are used in product development. After several iterations, the model described in this thesis was formulated. This model was coded and run as a computer simulation.

The conclusions drawn from the observations, literature, modeling and computer simulations were summarized as metaphors for thinking about prototyping and heuristics to guide the product developer.

PROJECT FOCUS ON PLASTIC PARTS

Plastic parts are an ideal subject for a prototype study. A plastic mold can cost between $50,000 and $100,000 and takes four to six months to make. The range is large and depends on the complexity of the mold. Complicated molds can cost much more and take much longer. If there is a problem with a mold, the repairs can cost thousands of dollars and take another month or two. These costs arise primarily because of the mold making process.

These are the typical (and greatly simplified) steps for making a mold:

1. Design mold.
2. Rough cut the part negative in each annealed steel mold half. Machined dimensions are close to the final desired dimensions.
3. Harden the steel with a heat treating process.
4. Grind or electric-discharge machine (EDM) the part negatives to the final dimensions.

The hardened steel of the mold and the complex features of most plastic parts are the primary contributors to the time and dollar costs. These same factors also make repairs expensive and time consuming. There is room to improve mold making by optimizing the production processes and flow, improving CAE links between process steps, and possibly by using new technologies like computer-controlled laser milling [Maho91] .

The high cost and time requirements of plastic mold making and the difficulty of mold repair create incentives to insure that the first design is correct. Building and testing prototypes is a good way to find problems early, i.e., buy "insurance" against possible future problems. Some of the processes i.e., buy "insurance" against possible future problems. Some of the processes that can be used to prototype plastic parts are listed in Table 2 and described in Appendix B.

Table 2. Some Processes Used to Prototype Plastic Parts

- Foam Core or Cardboard Assemblies
- CAD 2D
  3D Wireframe
  3D Solids
- CNC Machined Metal, Wood, or Some Other Material
- CNC Machined Plastic
- Finite Element Analysis
- Mold Flow Analysis
- Stereolithography
- Other SLA Type Layering Processes
- Spray Metal Molds
- "Engineering" Molds Made from Aluminum, Pre-hardened Steel or Some Other Material
- Single Cavity of a Production Mold
- Full Production Mold

OBSERVATIONS FROM PRACTICE AND IN THE LITERATURE

DISCUSSION OF PROTOTYPING OBSERVED AT KODAK

KAD: Possibly the World's Largest Job Shop

The Kodak Apparatus Division (KAD, pronounced "KayAdee") is housed in several large, interconnected buildings on Elmgrove Road in Rochester, NY. Under that one roof are some of the most advanced production technologies available in the world. It could be the world's largest job shop. The difference is that instead of a few milling machines and lathes, KAD has milling departments and turning departments that are the size of some factories. KAD also has a group that investigates new process technologies. They are a leader in using some of the new rapid prototyping processes such as stereolithography (SLA).

Two of the Business Units: Copiers and Cameras

On the same site are some of the business units that are responsible for actual products. Two groups that were major parts of this study were the copier group and the camera group. Both groups had a structure for their new product development process called "Phases and Gates". The phases are periods when things are happening in the product development cycle. A gate is a checkpoint where some deliverable is required before the next phase can begin. Examples of the deliverables include specifications, drawings, and prototypes. Figure 4 shows some typical parts for each department.

Both the copier group and the camera group had a "typical" sequence of prototyping. These are shown in node-network style in Figure 5.

Both groups started with 3D CAD representations, either wireframe or solids depending on the operator's skill. But after CAD, each group's sequence differs. There are several reasons for the different strategies which are based on the product development issues for each group:

The camera group uses CNC mills to machine their prototypes, and finally make the production mold.

- The camera group had a CNC model shop dedicated to their work. A liaison person between the engineers and the shop tracked the prototype work and expedited as required to finish the part.

- The camera group claimed CNC machining was required for their parts because stereolithography parts are brittle and can not be made within the required tolerances. Engineering molds are used for plastic spring parts.

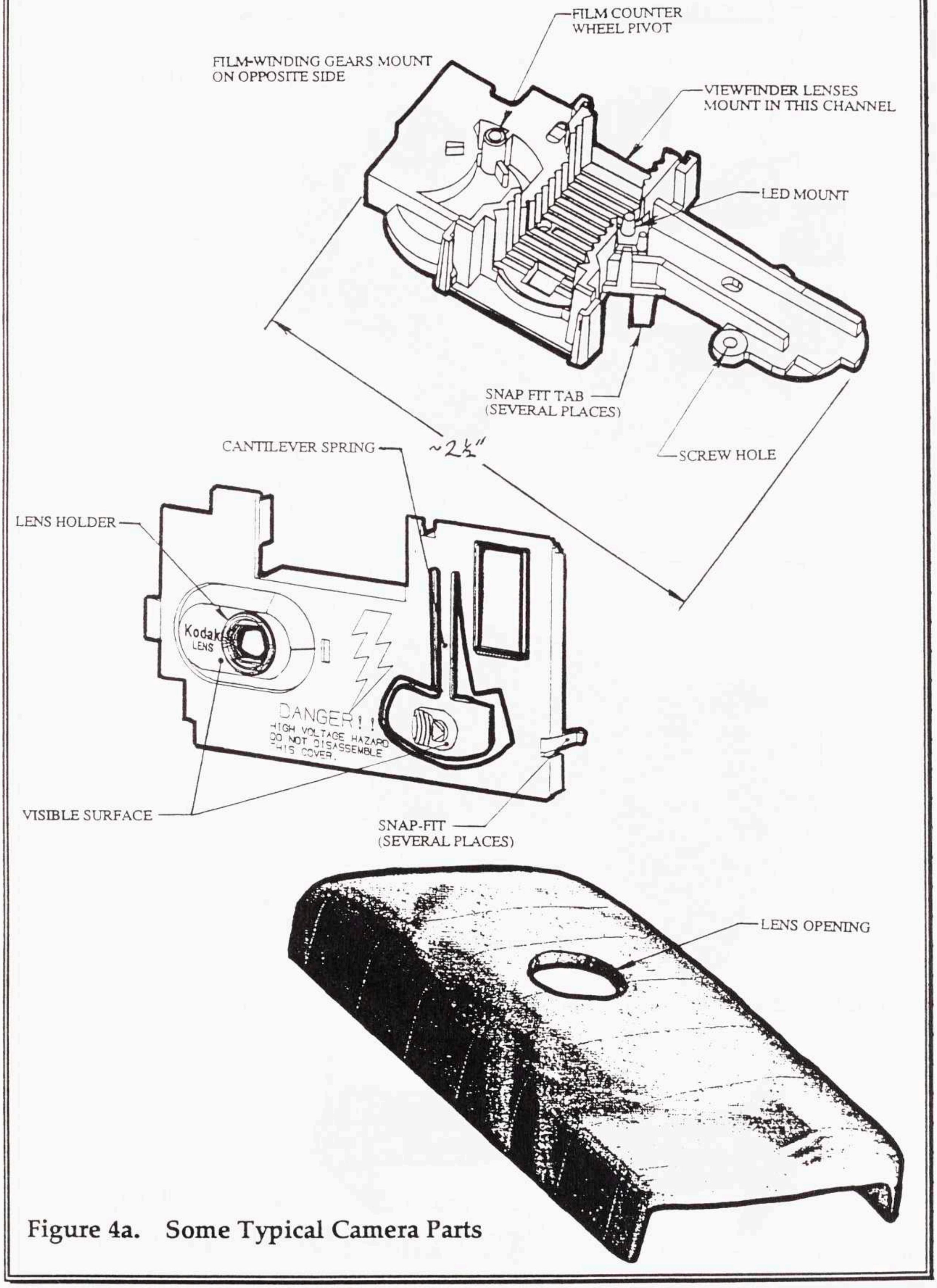


Figure 4a. Some Typical Camera Parts

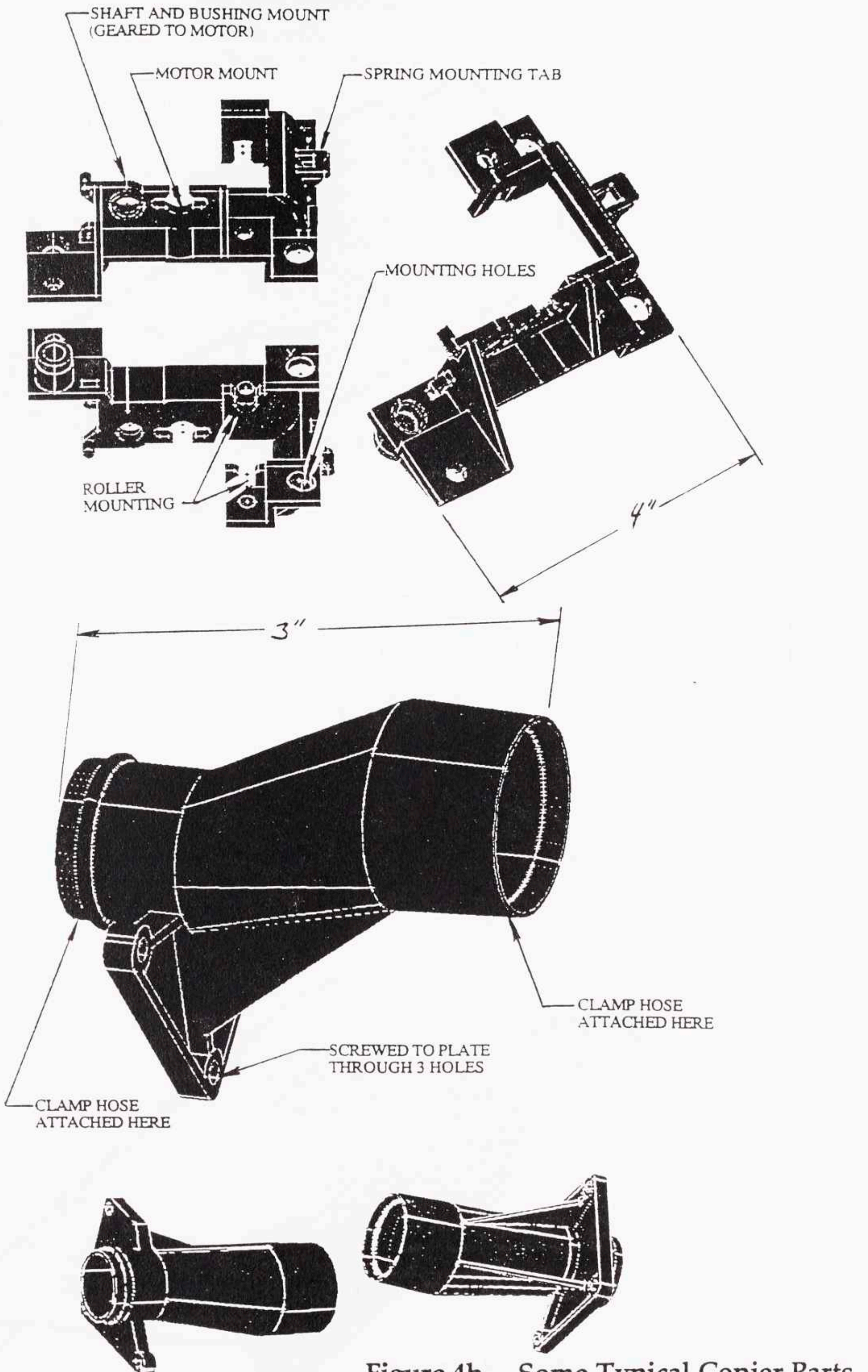


Figure 4b. Some Typical Copier Parts

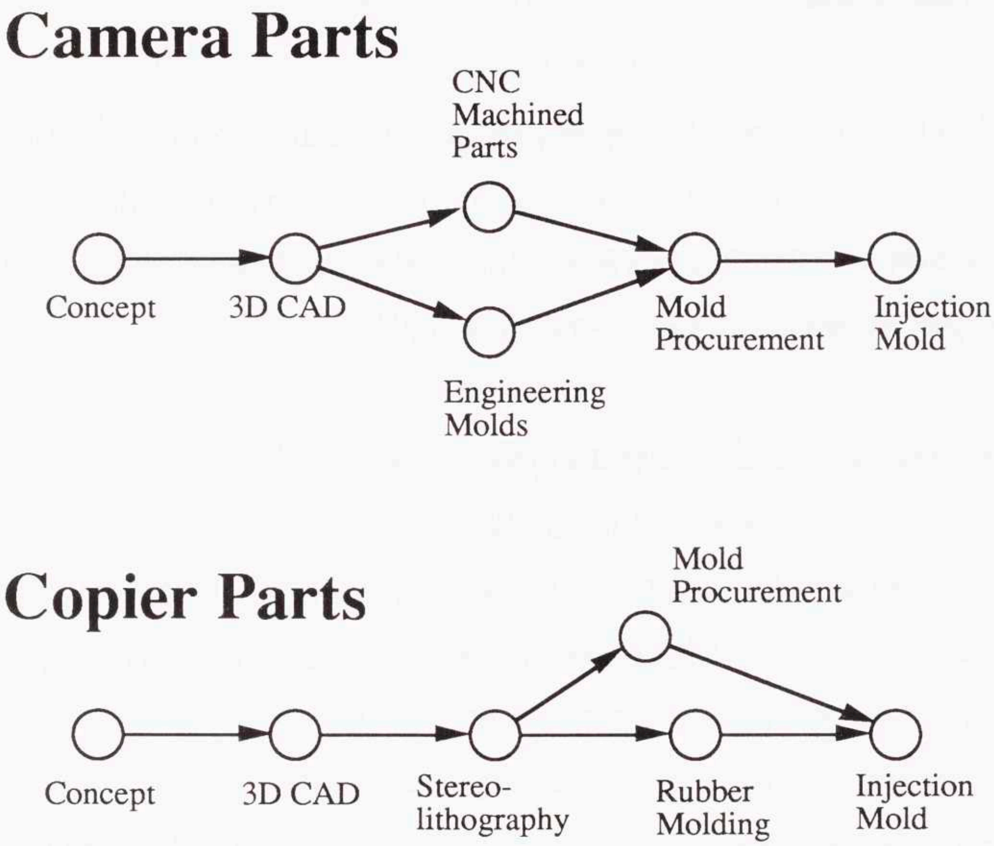


Figure 5. Two Typical Prototype Sequences

The copier group typically made a stereolithography part, then a rubber mold, and finally the production mold.

- The copier group is the premier user of KAD's stereolithography capabilities. It seemed as though every engineer had a SLA version of their part on their desk. Stereolithography was "*the*" process to use. In addition, the stereolithography part could be used as the master for a rubber mold.

- Several duplicates of a copier part prototype are required because there are several functional systems being tested at any given time. Each subsystem group (eg. software, toner, etc.) needs an almost complete copier system to optimize their portion.

I would estimate that the individual parts of the two departments were at similar levels of complexity. But I think factors that contribute to the different styles of prototyping include the complexity of the final products, the product development environments, the resources easily available, and the traditions of past practice.

Anecdotal Examples of Good Prototyping Practice

During my interviews of designers, engineers, and model makers, some stories stand out as interesting lessons in prototype practice:

- <u>The value of the production group making the prototypes.</u>
  During development of one of the recent copiers, a person in the mold-making area was responsible for having all the prototypes made for the designers regardless of the process or vendor used. This person had a good understanding of the production molding process. He was able to make producibility suggestions to the designers because he saw all the parts the first time they were to be prototyped.

- The value of a physical prototype.

  The ease with which one can manipulate and test an actual part is much greater than that in a typical 3D CAD system. Here is an example:

  The part was an air pressure adjuster for the vacuum line in a copier paper handling system. The parts consisted of two sleeves, one inside the other. Each sleeve had a slot cut along part of its circumference. The vacuum adjustment was made by rotating the outer sleeve to change the amount the two slots overlapped. With the stereolithography part, the designer quickly noticed that when the outer sleeve was rotated closed, the front of the outer slot overlapped the back of the inner slot creating an open gap. The part had an aliasing problem. This could have been noticed by manipulating the CAD parts, but it was not. There is a large difference between physical parts and CAD for this type of manipulation and visualization problem.

  I do not think examples like the one above occur because CAD operators are necessarily sloppy, I believe that any CAD operator will miss some things because of the nature of the CAD interface. Each operator stops checking the part when they believe they have reached a point of limited return. Hopefully, occurrences like the example above will become less common as CAD tools improve and part representations are easier to play around with. Better CAD interfaces should encourage more part experimentation.

- <u>How fixing one problem can cause other unexpected problems.</u>
  "If it's not one thing, it's another." This is an example where fixing one problem created problems of an entirely different type:

  The plate is part of the paper path in the copier's document feeder. Prototypes were made with rubber molded parts. The production parts were injection molded from a different material. When the parts were installed in the document feeder, the engineers noticed that static electricity interfered with the paper handling, and the part had wear problems because of the abrasive edges of paper. To fix the problems, the part was nickel plated. The nickel was tough to resist wear and conductive to bleed off the static charges. But the nickel plating created other problems. First there were burrs created by the plating process; the paper hung on the burrs. Second the plating changed the mounting hole diameters, which created fit problems in the assembly. The plating process had to be improved and the mold had to be changed to correct these new problems.

- <u>Multi-functional parts can be prototyped incrementally.</u>
  Many parts of a camera perform more than one function. One engineer built his prototype function by function. The first prototype tested one function of the part. Then the second function was added to the first prototype. This was an effective way to reduce total prototyping time and cost.

I collected many ideas and facts about prototyping at Kodak. To help organize my thoughts on the subject, I gathered my observations and made a KJ diagram on the question "What characterizes the prototyping process?" The KJ Method, which is registered with the Japanese Patent Office, involves making and grouping labels of qualitative data. The groups form a meaningful body of information that can provide a basis for judgement [Kawakita90]. The result is shown in Figure 6 and Appendix C. One "reads" the KJ by the group titles and the relationships between the groups. At the highest level, this KJ shows how engineers' desire to prototype, organizational constraints, and field repair costs make lead time reduction and risk management important. This is shown in the lower half of Figure 6, where three groups point into one. These then lead to the large group at the top that summarizes the need for sound strategy and knowledgeable engineers to do effective product development. The lower level titles and original comments are also interesting to read. Because of space constraints in this paper, the lowest level points are letter coded in Figure 6 and listed along with the group titles Appendix C.

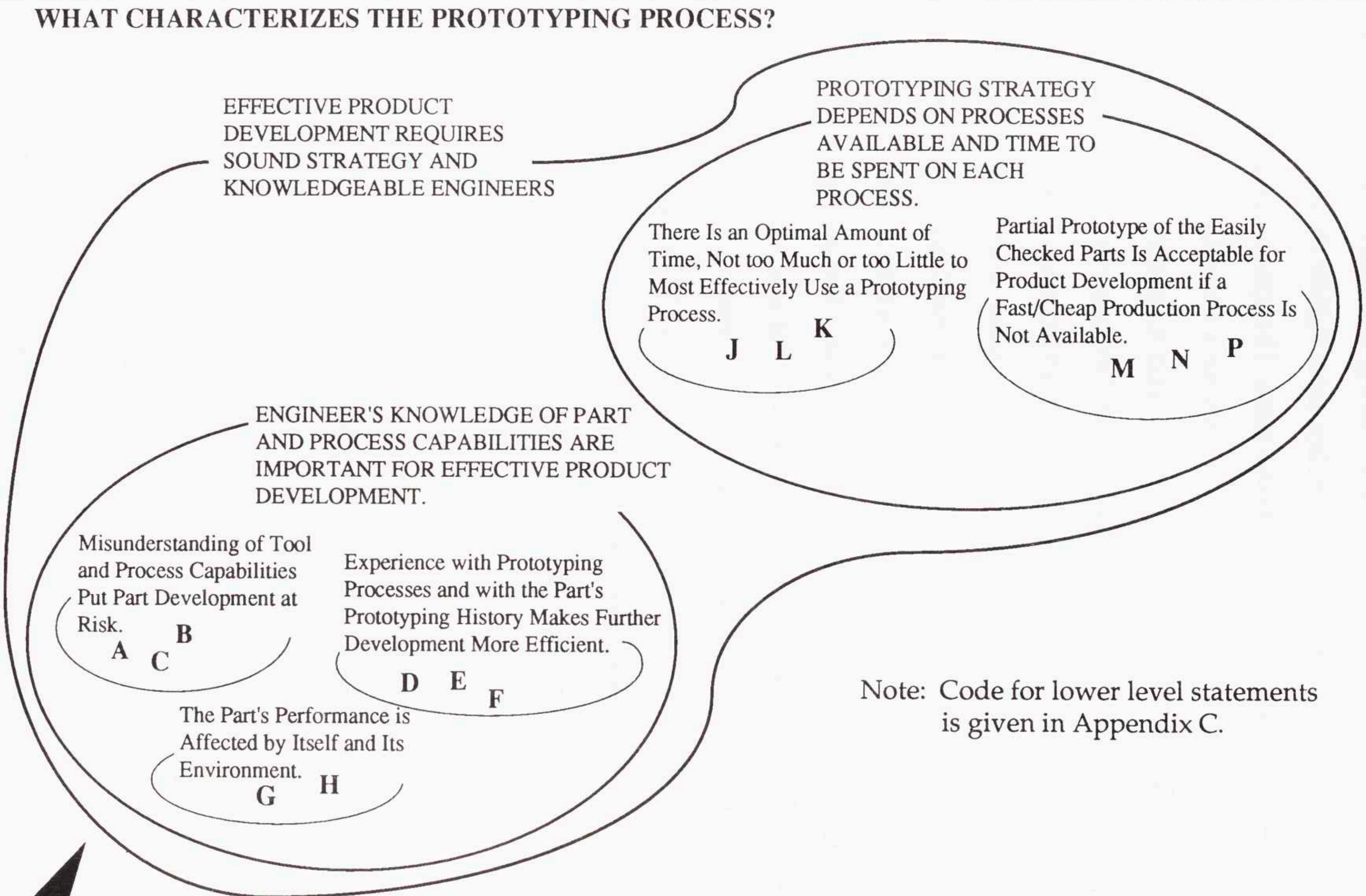


Figure 6. KJ Diagram of Prototyping Observations (top half)

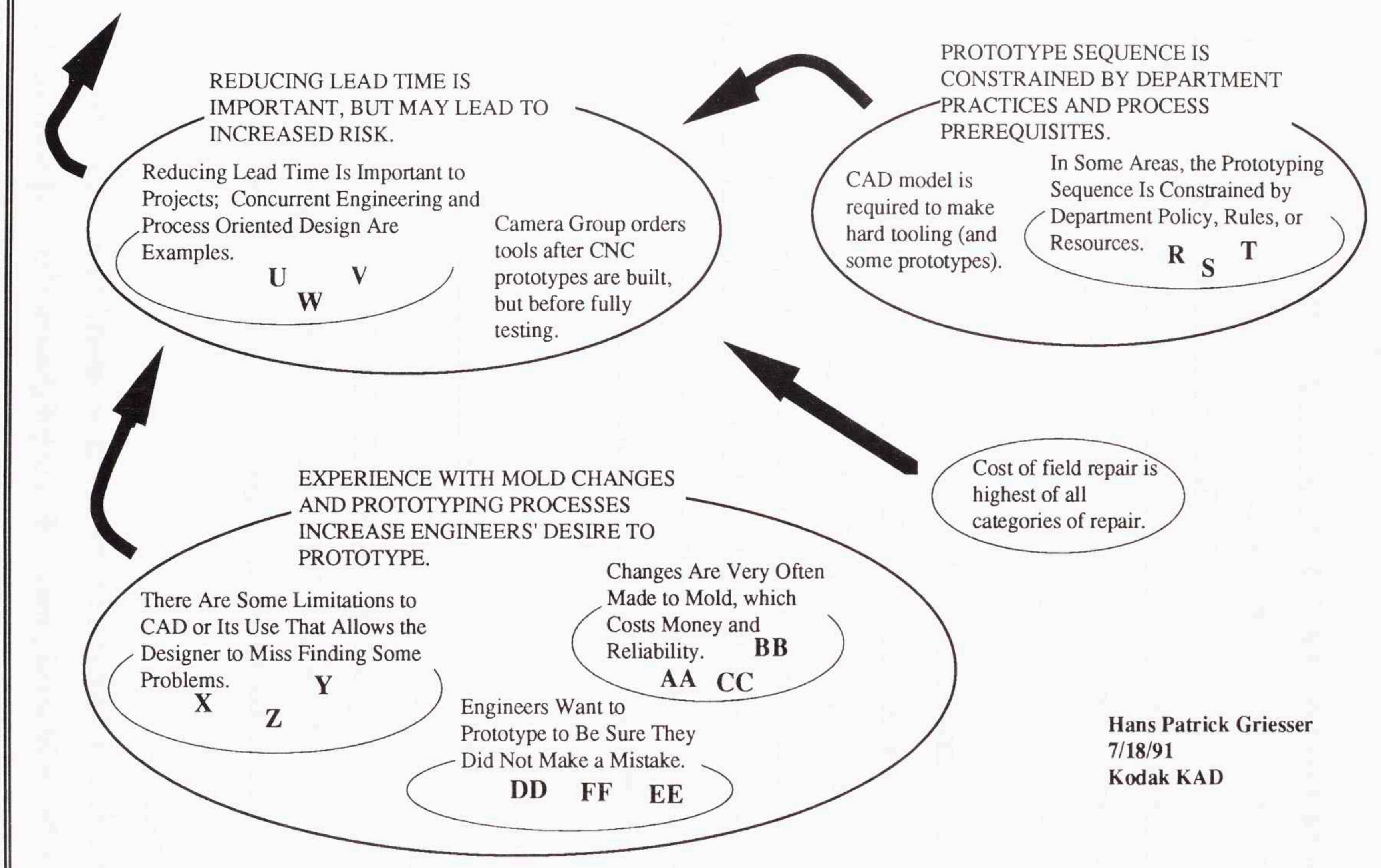


Figure 6. KJ Diagram of Prototyping Observations (bottom half)

REVIEW OF EXISTING PROTOTYPING LITERATURE

The difficulty in doing a literature search on prototyping is that the key word, "prototyping" yields thousand of entries on particular prototype products and processes but very little on prototyping strategy. Regardless, the following is a short summary of some of the current thinking about prototyping and how it relates to product development.

Building on Wall's Work

Much of the work in this thesis is an extension of Matt Wall's thesis, the result of his LFM internship at Kodak. Wall describes a framework for determining which process is best to use for a particular type of prototype, i.e., one for a certain purpose [Wall91]. In the network model described in Figure 1, Wall's framework determines the process to use for an individual node. The model in my thesis addresses the entire network.

Prototyping Strategies

There are three groups of strategies in the literature that describe how to use prototypes.

- General strategies that provide broad guidance
- Detailed strategies in which every step is carefully specified
- Promotional strategies that suggest using a particular process

General Strategies

The first class of strategies are those that give general instructions. One recommendation in using prototypes is to progress to richer representations. Move through a sequence of 2-D Models, Non-Function 3-D Models, Functional Prototypes, User Test Models, and Organization/System Models [Leonard-Barton91]. Another idea is to use the product as a prototype. Freeze the design early and get it to the market. Customer feedback will determine the next iteration. Casio uses this strategy for their large line of wrist watches [Smith92]. These general strategies could work for a variety of products.

There are some other general suggestions to keep in mind. For instance, not every product needs to be on the rapid development cycle [Smith92]. A design department's work load can be balanced by prioritizing the work. Also, inappropriate tools can severely constrain performance. Old solutions can be cloned and creativity stifled which may result in sub-optimal designs [Murotake91].

Detailed Strategies

In the literature are some very detailed descriptions of how products should be developed and how prototypes can be used.

Clausing describes a very detailed generic product development plan. The prototype required for each development stage is specified. The number of prototypes are limited to a maximum of four. This is to avoid "Hardware Swamp" where a large number of highly overlapped prototype iterations leave little time for improvement solutions. Each prototype must be planned to make the maximum contribution to optimization [Clausing91].

Buttrell also has a detailed product development plan which includes check lists for each stage. In his plan, working prototypes are made after the product definition and the preliminary profit plan are complete. This focuses the product developer's thoughts on the product and allows the decision to continue or quit the project to be economically made. There are also production prototypes, which need a "shepherd" to get through the system, and into field testing [Buttrell84].

In my opinion, both of these systems lack flexibility and may be too detailed to address every product line. They must be customized to fit a particular organization or product type. Their value is that it makes the design process explicit allowing the group to evaluate and improve their product development process.

Promotion Strategies

There are many articles written that promote one or another technology for prototyping. These articles frequently include a product development flow map where one of the elements is the prototyping technology being promoted. Comments like stereolithography "looks to become a standard engineering tool in the 1990's" [Deitz90] are common .

Some typical examples of articles that promote a technology include one that promotes automatic mesh generation algorithms, which help determine the elements of a part for finite element analysis, for concurrent engineering [Teague91] and another that claims that simulations of product interface help designs develop faster than physical prototyping of the interface [Gardner92].

A similar type of article promotes a specific design process offered by a vendor. For instance, one vendor offers to integrate simulations and prototypes in parallel with the concept generation, design and manufacturing phases of product development, all part of the team concept [Trapp91]. Another article compares the many different methodologies available and their capabilities. Processes such as Boothroyd-Dewhurst DFA, Taguchi Methods, or FMEA have different value in different areas [Stoll88].

Value of Prototypes

The value of prototypes are mentioned in several articles: Points listed include the following:

- Selling the concept
- Spotting problems
- Verifying and optimizing designs
- Getting a fast start for tool making [Miller91]

Benefits from using prototypes include:

- Fewer design changes
- Continuous feedback loops
- Lower scrap rates
- Less warranty problems
- Increased market share and profitability [Gardner92]
- Lower mold quotes because the vendor has a better understanding of the part.

A prototype is used as leverage to direct product development and map product requirements to market [Wheelwright89]. It can also direct product development because it captures the current understanding of the part and reveals the next set of questions to be answered [Leonard-Barton91].

For the reasons above, several large, product-oriented companies are doing "prolific prototyping" as part of their product-development strategies [Schrage91].

Prototypes for Communication

One recurring theme in the literature was the value of prototypes for communication between all the groups involved in the manufacturing enterprise.

A prototype, particularly a physical one, is something that everyone can relate to and experience. It is a central entity that will translate the different languages of each group into one that all can understand. Teams discuss many different things, but different disciplines have different meanings for terms like time or quality. A model or drawing can be the link between the disciplines [Powell89].

This property of a prototype is called Boundary Spanning. It is significant because models are neutral, visible, and accessible symbols of the final product [Leonard-Barton91]. They become the focus of arguments (design discussions) because they stand between proof-of-concept and the end product [Mogavero82]. This quality to engender communication between disciplines is very important because groups still tend to communicate more within than across departments [Leonard-Barton91]. Prototypes must be shared with everyone especially, of course, with the customer [Schrage91].

Prototyping Models and Strategy Tools

Frequently discussions of prototyping strategies are part of a larger product development strategy.

Product Development with Templating

An interesting way to think about product development is with templating. The list of needs and concerns for a new product is called a template. This differs from a written product specification in that a template is the mental model for the product and it includes needs that are not always easy to specify. A new design is judged by how well it matches the template. People learn by trying to match their template with a proposed solution and looking for any misfits [Tyre91].

The catch is that a complete template is difficult to specify because people only recognize the misfits and ignore the qualities that are already right [Tyre91]. Decision making can also be done by changing the template, i.e., the constraints to fit a particular alternative [Frischmuth69].

Prototyping Methodology

There are several methods to determine the tasks and the order of tasks to do in product development.

- Do a financial analysis that includes the economic cost of time-to-market. For instance, studies show that this can lead to design decisions other than DFM rules would suggest [Ulrich91]. The same could be said for prototyping as well.

- Do an economic analysis include economic advantages such as improved product quality or increased flexibility. This is particularly useful for new technologies [Kutay90]. The difficulty is that these advantages can be hard to quantify.

- Use Design Structure Matrices to organize product tasks. The matrix elements consist of probabilities that describe the level of coupling between tasks. The methodology helps designers organize the tasks to reduce the total development time [Smith91].

- Implement an explicit design strategy by using the confidence in design parameters to trade off performance against those design parameters [Antonsson92].

Prototyping in Other Domains

This section compares prototyping in product design with some other domains.

Developing Information Technology Systems

Another field that is increasingly using prototypes is the development of computerized information management systems. Great productivity gains are expected from using more prototypes. The design problems of information systems are similar to those of products:

- The developers have difficulty in pre-specifying exact requirements.
- Documentation is inadequate to communicate the intricacies of product.
- In general, miscommunication is endemic [Boar84].

Prototypes such as mockups in the hallway, slide shows, or graphical software tools help developers get a more complete specification, prove technical feasibility and determine the best implementation [Dickerson88]. Developers also use expert system shells, which identify expected problems, to lower development risks in designing network management systems [Noren88]. In general, developers desire better systems that better represent the look and feel of the final product [Dickerson88]. Literature describing information system development life cycles now include prototyping requirements [Li90] and the technical facilities required to prototype [Ince87].

Determining Product Inspection Points

There are some similarities between prototyping and product inspection. They both involve incurring a cost in an attempt to learn something. A test (or a prototype) must be powerful enough that the outcome affects the correct decision and must be inexpensive enough that the resulting decision gives a net gain.

Methodologies that determine the optimal sequence of test points in a production line use dynamic programming, simulations, or non-linear programming [Raz86]. One of the dynamic programs was used to test heuristics to optimize inspection of a serial production line [Peters84]. The heuristics that the program supported include the following:

- "Inspect prior to processing operations that may render future detection of non-conforming unit difficult and costly." In the prototyping domain, this may be stated as, "Prototype before committing to a critical design decision that may render future improvements difficult to make." For instance, don't show the design to a dictatorial boss until it is thoroughly tested. A decision by him based on a poor prototype could stop a promising project (and career).

- "Inspect after processing operations that generate a high proportion of non-conforming units." In the prototyping domain, this may be stated as, "Prototype after making a lot of design decisions because each decision increases the probability of a new design problem due to coupling."[1]

Peters's model did not support the heuristic: "Inspect prior to costly processing operations," which in prototyping would be "Prototype prior to costly design tasks." However, the authors did warn that the results depended on specific parameter values and assumptions [Peters84].

Prototyping Consulting Solutions

A friend of mine is an in-house consultant for a large computer company. She says they have begun to use a prototyping approach with their client groups. In her words, "The Big Bang Theory just doesn't work". That is, they don't deliver one grand solution (typically a thick report that no one wants to read) at the end of a study. Instead, they develop the solution and share it with the client as they go along. In this way, they improve client communications and increase customer satisfaction. Periodic feedback from the client is invaluable in shaping the final product [Sutton92].

---

[1] This type of coupling is discussed later in the prototyping model section.

Determining Financial Trades

There is a financial trading analogy for prototyping. Financial trading involves cost, risk and returns. There may be some financial trading algorithms that can be translated into the prototyping domain. For instance, "Buy low, sell high" can be interpreted as, "Make quick and dirty prototypes that teach you a lot." But I did not find any literature on optimization programs for financial trading that I could apply to prototyping.

A MODEL OF THE EFFECTIVENESS OF A PROTOTYPE SEQUENCE

To gain insight and a better understanding of how prototypes are used in the product development process, my observations of practice were condensed, refined, and summarized into a model. The model gives a framework to study and consider prototyping strategies.

The inputs to the model are the following:

- Numerical representations of the product developer's concern about the part attributes
- The set of prototyping processes that will be used and how much the prototypes will be studied

The outputs of the model are the following:

- The time to completion
- The amount the concerns have been addressed

These inputs and outputs can be used in several ways:

- Evaluate a particular set of prototyping processes
- Find the optimal set of prototyping processes for a given starting set of concerns
- Test heuristics that specify prototyping practice.

These are explained in greater detail in this section.

THE MODEL'S ASSUMPTIONS

The assumptions made in the development of the prototype sequencing model are summarized in Table 3. They are describe in detail below.

Table 3. The Model's Assumptions

- The state of a part's development can be described at any point by the concern levels of the part's attributes.
- A prototyping process can be represented by:
  - Build Time
  - Effective Study Time
  - Ability to Address the Attributes of Concern.
- The order of prototyping processes does not affect their abilities or the prototyping sequence outcome.
- Only one process may be done at at a time.
- Something is always being done.

A Part's Development State Is Described by Attribute Concern Levels.

The model assumes that any point in a part's development can be described by the levels of concern for particular attributes of the part. These attributes can include dimensions such as geometry, material properties, appearance, and any other property one might consider appropriate. The concern levels start at some arbitrary value (eg. 4) and are reduced to zero when the part is in production and all problems are solved. The concern levels are recorded as elements in a vector.

Product development is a process of reducing the concern levels, i.e., learning about the part's problems and correcting them. The product development can be visualized by tracing the path of the vector through the attribute concern space. A graphical description of a vector's progress in a 3-attribute dimensioned space is shown in Figure 7.

The concern level is basically an arbitrary weighing set by the designer to each attribute. The relative concern levels between the attributes are more important than the absolute values. The units are not named. But if naming the units is required, I suggest "problons".

The concern levels do not always have to be reduced. Sometimes addressing one concern can create problems in another dimension. For example, adding nickel plate to a plastic part to eliminate static electricity will increase the part thickness and possibly create assembly fit problems. This is called coupling.

Processes Characterized by Three Dimensions

In the model, a prototyping process is characterized by the three dimensions:

- Build Time
- Ability to Address Concerns
- Effective Study Time

The dimensions are described below and shown graphically in Figure 8.

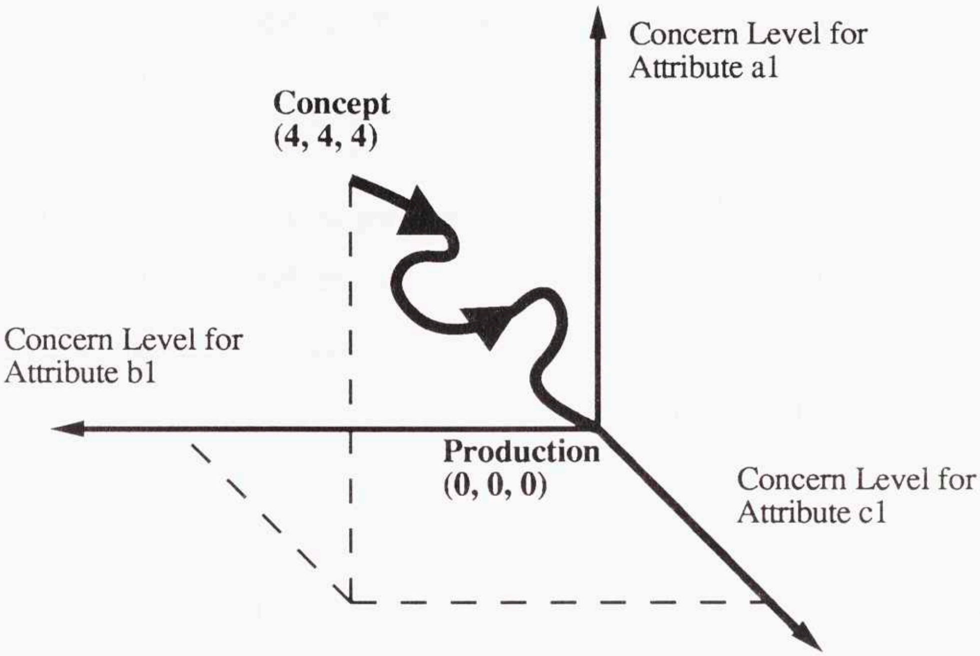


Figure 7. **Concern Vector Path through Concern Attribute Space**
The concern levels for all the part's attributes begin high. As the product development progresses the concern levels rise and fall until production is well underway, all problems are solved, and all the concern levels are zero.

Build Time

While a prototype is being built, the model assumes no learning is done and no concern levels are reduced. The "BUILD TIME" is simply the amount of time required to make the prototype before it can be tested.

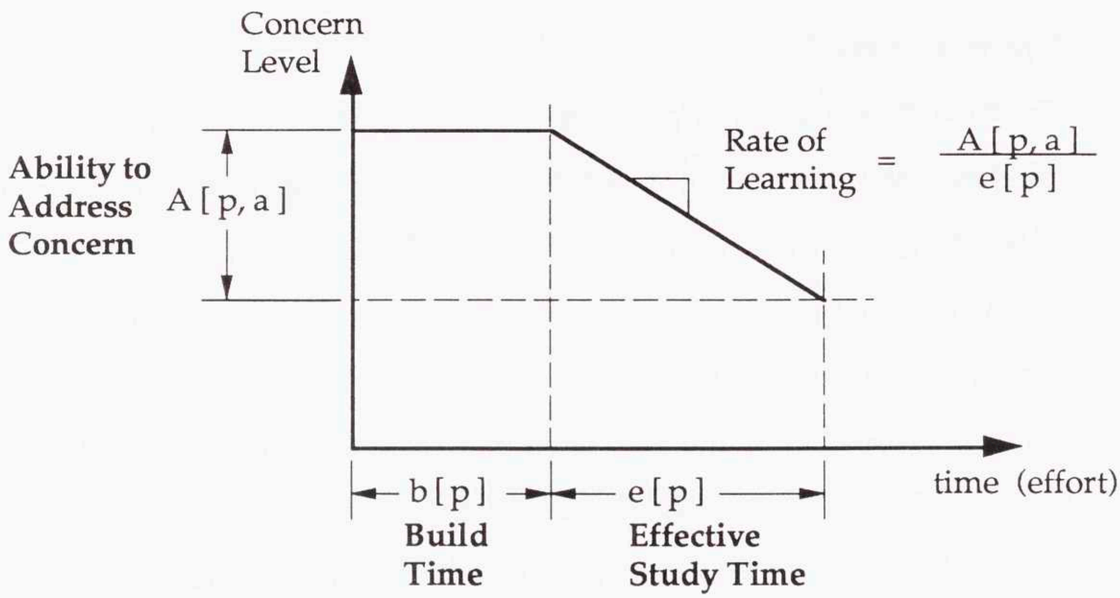


Figure 8. Prototyping Process Characteristics
The variables shown for the three characteristics are for the formal model explanation in a following section.

Ability to Address a Concern

Some prototype processes are better at testing some attribute concerns than others. This difference between processes is captured in the process dimension "ABILITY TO ADDRESS CONCERN." The ability is represented as the percentage of problems a process can address in a given category. For example, stereolithography may be able to address all the "Geometric Fit" concerns but only a fraction of the "Strength" concerns. A process has a separate ability rating for each of the attribute concerns of interest.

Effective Study Time

The final dimension of a prototype process is the amount of effort that must be used to gain the maximum utility, i.e, the maximum amount of learning, from that prototype. The "EFFECTIVE STUDY TIME" is a measure of that effort. If only a fraction of that time is used, only a proportionate amount of the possible concern level reduction is realized. However, if the time used is greater than the EFFECTIVE STUDY TIME, no extra learning is done. The amount of learning is limited by the "ABILITY TO ADDRESS CONCERN" percentage. This is a mechanism to capture the decreasing marginal return of using a prototype.

The Effective Study Time combined with the Ability to Address Concerns and the initial concern level give the rate of learning available for a prototyping process. This is the slope of the learning (concern reduction) line in Figure 8.

Sequencing Order Does Not Affect a Process's Abilities or the Sequence Outcome

Because of the simplifications made to characterize a part's development and a process's abilities, the order of prototyping steps is not relevant. The order will affect the learning rate because it is dependent on the initial concern level. But, because each process reduces a percentage of that concern level, the final outcome of a sequence is the same. The percentages are multiplied, and so prototyping in this model is commutative.

This assumption does not match reality. For example, some processes require having one type of prototype before the process can be used. Rubber molds require a master part, and FEA requires a CAD model. To match this reality, these additional constraints must be added to the model. The constraints were not include here because they were not required for the first set of insights.

Only One Process Can Be Done at a Time

Because of the limits of my calculation abilities and to eliminate some coupling effects, the model constrains the prototype sequence to one process at a time. This means that all the learning from one prototype must be finished before the next prototype can be built. And, there is no learning being done while a prototype is being built.

If a part has features that can be decoupled and tested simultaneously with separate prototypes, separate model runs can be done for each feature.

Something is Always Being Done

The model assumes that at any given time, a prototype is either being built or used for testing and learning. This implies that effort and calendar time are synonymous. They are used that way throughout this thesis.

THE MODEL'S OPERATION

This section describes in detail how the model works and what calculations are done. The variables used in the formulas are listed in Table 4.

Table 4. Variables Used in this Thesis

a, a'	Numbers that correspond to attributes, i.e., concern categories.
A	Ability Matrix.
A[p, a]	Element in **A** that corresponds to the ability of process p to address attribute a. $0 \leq$ **A**[p, a] ≤ 1.
b	Build Time Vector.
b [p]	Element in **b** that corresponds to the time to make a prototype using process p.
c	Concern Vector.
c [a]	Element in **c** that corresponds to the concern level of attribute a.
c [a]$_0$	Original value of **c** [a] before learning from the next prototype.
c [a']	Element in **c** that corresponds to the concern level of attribute a'.
e	Effective Study Time Vector.
e [p]	Element in **e** that corresponds to the time to get all the learning possible from a prototype made with process p.
L	Coupling Matrix.
L [a, a']	Element in **L** that corresponds to the amount of coupling, the probability of a problem arising in attribute a' due to an improvement in attribute a.
n	Number of times the process, p, is repeated.
p, p', p"	Numbers that correspond to prototyping processes.
t_s	Development Time or Study Time, i.e., actual time spent trying to learn from the prototype.
T	Total Development Time for a sequence of prototype processes.
T_0	Original Total Development Time before learning from the next prototype.
v	Threshold Value below which Concern Levels are considered negligible, i.e., development of the attribute or learning about the attribute is complete.

Inputs and Outputs

The inputs to the model are the concern levels for the part attributes. These are recorded in the Concern Vector, **c**. Each element of the vector represents the concern level for a particular attribute. Table 5 shows two possible sets of attributes for the Concern Vector. The length of the Concern Vector is equal to the number of attributes.

Table 5. Possible Concern Vector Elements

A possible 10-element concern vector:

c = { Stiffness/Strength from Material,
Stiffness/Strength from Geometry,
Function from Material,
Function from Geometry,
Life of Part,
Ability to Assemble Part in Product,
Function with Mating Parts from Geometry,
producibility,
Marketability,
Serviceability }

A possible 4-element concern vector:

c = { Function from Geometry,
Function from Material,
Marketability,
Manufacturability }

The outputs of the model are the following:

- the sequence of prototype processes which will reduce all the concern levels below some threshold level in the least amount of time,
- the time required to complete this sequence, and
- the final concern levels.

Alternatively, the model can be used to estimate the time to completion and the final concern levels for a given initial Concern Vector and a given process sequence. Or the model can be used to test heuristics for selecting prototype process against the optimal sequence.

Reduction of Concern Levels

The model works by using the known prototype process characteristics to reduce the concern levels in **c**. As described earlier, a process is described by three characteristics: Build Time, Effective Study Time, and Ability to Address Concerns. In the model, the characteristics for all the processes are recorded in a vector or matrix for look-up purposes. The values in the vectors and matrices were estimated from my observations at KAD, especially the part histories in Appendix A. To generate the values, I categorized all the problems encountered and the prototype processes that revealed the problems. This information was the basis for each process's characteristic values. The vectors and matrices are easily changed to match the process capabilities of a particular development environment. The process characteristic vectors and matrices are described below:

- <u>Build Time Vector, **b**</u>

  The length of this vector is equal to the number of processes considered in the model run. Each element corresponds to the time required to make a part using a particular prototyping process. A sample Build Time Vector is shown in Figure 9.

b =	[2		Mold Flow
	2		FEA
	2		SLA
	5		CNC/Plastic
	8		Rubber Mold
	60		Engineering Mold
	100	]	Production Mold

Figure 9. Sample Build Time Vector
Each element represents an approximation of the time (days) to build a prototype using the corresponding process.

- Ability Matrix, **A**

  One dimension of the Ability Matrix is the number of processes considered; the other is the number of Concerns in **c**. The matrix is a grid where each element corresponds to the ability that a particular process has to test a particular attribute. Each element is between zero and one:

  $$0 \leq \mathbf{A}[\,p, a\,] \leq 1$$

  1 = Ability to fully test attribute concern.

  0 = No ability to test attribute concern.

  A sample Ability Matrix is shown in Figure 10.

- Effective Study Time Vector, **e**

  The length of the Effective Study Time Vector is equal to the number of processes. Each element corresponds to the time required to completely test a part using a particular prototyping process. A sample Effective Study Time Vector is shown in Figure 11.

Attributes: / **Processes:**

A =

Processes:	Stiffness/Strength - Material	Stiffness/Strength - Geometry	Function from Material	Function from Geometry	Life of Part	Ability to Assemble	Function with Mating Parts	Producibility	Marketability	Serviceability
CAD	0.	.3	0.	.8	0.	.8	.8	.4	.2	.1
FEA	0.	.9	0.	0.	0.	0.	0.	0.	0.	0.
Mold Flow	.4	.3	.2	0.	0.	0.	0.	.7	.1	0.
Stereolithography	0.	.1	0.	.4	0.	.8	0.	0.	0.	.8
CNC of Metal	0.	.1	0.	.7	0.	.9	.8	.3	.3	.8
CNC of Plastic	.5	.5	.5	.9	.2	.9	.9	.3	.8	0.
Rubber Mold	.5	.8	.5	.9	.3	.9	.9	.8	.9	.9
Spray Metal	.7	.9	.7	.9	.5	1.	1.	1.	1.	1.
Engineering Mold	1.	1.	1.	1.	1.	1.	1.	1.	1.	1.
Production Mold	1.	1.	1.	1.	1.	1.	1.	1.	1.	1.

Figure 10. **Sample Ability Matrix**
Each element represents the approximate percentage of concerns that the corresponding process can address for the corresponding attribute.

e = [

2	Mold Flow
2	FEA
2	SLA
5	CNC/Plastic
8	Rubber Mold
15	Engineering Mold
40]	Production Mold

Figure 11. **Sample Effective Study Time Vector**
Each element represents an approximation of the amount of effort (days) that must be used to gain the maximum utility, i.e, the maximum amount of learning, from a prototype made with the corresponding process.

Coupling Effects

To account for possible coupling effects between attributes, an Attribute Coupling Matrix, **L**, is defined. Coupling occurs when improving one attribute creates problems in another attribute. This matrix is square with the number of elements on each side equal to the number of Concerns in **c**. Each element corresponds to the probability that making an improvement in one attribute, a, will have a negative effect on another attribute, a'. This is recorded as an increase in the affected concern level, a'. The absolute value of the matrix element, L[a, a'], is proportional to that probability. The elements are made negative for the calculation reasons described below. A sample Attribute Coupling Matrix is shown in Figure 12.

Only negative effects are considered in the Coupling Matrix. This keeps the model conservative. Many attributes are coupled in some way, but because only negative effects are considered in the model, coupling can not lower the concern levels; coupling can only raise the concern levels. Any positive coupling effects, ones that would lower the concern levels, are implicit in the Ability Matrix. They are taken into account in the inherent structure of the model. If a prototyping process can address more than one attribute, those attributes can be considered positively coupled. To include this coupling in the Coupling Matrix would be redundant.

Although the matrix elements represent probabilities, they are treated as deterministic because raising the concern level is a conservative thing to do. The elements could be treated as probabilities instead. This would add a stochastic element to the model which would more accurately reflect prototyping practice. However, when I mention the "probability of coupling" in the following descriptions, I treat it as deterministic.

Attributes, a:

L =

Attributes, a':	Stiffness/Strength - Material	Stiffness/Strength - Geometry	Function from Material	Function from Geometry	Life of Part	Ability to Assemble	Function with Mating Parts	Producibility	Marketability	Serviceability
Stiff/Strength - Material	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
Stiff/Strength - Geometry	-.3	0.	-.1	-.7	0.	-.5	-.2	0.	0.	-.1
Function from Material	-.1	0.	0.	0.	0.	0.	0.	0.	0.	0.
Function from Geometry	-.6	-.7	-.2	0.	0.	-.4	0.	-.3	0.	0.
Life of Part	-.2	-.2	-.7	-.2	0.	-.1	-.2	0.	0.	-.2
Ability to Assemble	0.	-.7	0.	-.7	0.	0.	-.3	0.	-.4	0.
Function w/ Mating Parts	-.8	-.8	-.8	-.8	-.2	-.6	0.	0.	0.	-.2
Producibility	0.	-.4	-.2	-.7	-.1	0.	-.7	0.	-.8	-.1
Marketability	-.1	-.6	-.1	-.7	0.	-.7	0.	-.8	0.	-.1
Serviceability	0.	-.2	0.	-.2	0.	0.	-.6	-.1	-.2	0.

Figure 12. Sample Coupling Matrix
Each element represents the probability of a problem arising in attribute a' due to an improvement in attribute a. The numbers are negative for calculation reasons; the coupling rises the concern levels.

Exercising the Model

This section discusses the actual mechanisms in the model that change the concern levels by prototyping. This is the "engine" of the model. It can be used in the various ways described in the previous *Inputs and Outputs* section. Here is how the engine works:

The model takes the initial set of concern levels and reduces them according to the prototyping process used. The development time is incremented by the amount of time required to build and test the prototype. The concern levels are then raised by the coupling factor. The entire process is repeated until the all the concern levels are reduced to zero. This process is shown graphically in Figure 13. The details of this process are given below.

1. Select initial concern levels and place them in **c**.

2. Select a prototype process, p.

3. Increment the total development time, T, by the build time value in **b** [p] that corresponds to prototype process p. Also increment, T, by the development time, t_s, used to test and study the prototype. The development time (or study time, i.e, the time spent trying to learn from the prototype), t_s, is an input to the model.

$$T = T_0 + \mathbf{b}[\,p\,] + t_s \tag{1}$$

4. Reduce the concern levels in **c** according to one of following scenarios:

   - If the study time, t_s, is greater than or equal to the processes Effective Study Time, **e** [p], then reduce each attribute, **c** [a], by the proportion that corresponds to the Ability Matrix element for that process and that attribute, **A**[p, a].

     If $t_s \geq \mathbf{e}[p]$, then

     $$\mathbf{c}[a] = \mathbf{c}[a]_0 \bullet (1 - \mathbf{A}[p, a]) \quad \text{for all a.} \qquad (2a)$$

   - If the study time, t_s, is less than processes Effective Study Time, **e** [p], then reduce each attribute, **c** [a], by the fraction of the proportion that corresponds to the Ability Matrix element for that process and that attribute, **A** [p, a]. The fraction is determined by the ratio of the actual study time to the Effective Study Time, t_s / **e** [p].

     If $t_s < \mathbf{e}[p]$, then

     $$\mathbf{c}[a] = \mathbf{c}[a]_0 \bullet (1 - \mathbf{A}[p, a] \bullet (t_s / \mathbf{e}[p])) \text{ for all a.} \qquad (2b)$$

5. Increase the concern levels to account for any coupling effects caused by the learning and improvements just done. If an improvement was made to an attribute, a, that is coupled to another attribute, a', increase the concern level of a' by the product of the change in concern level and the coupling coefficient in **P**.

$$\mathbf{c}[a'] = (\mathbf{c}[a]_0 - \mathbf{c}[a]) \bullet \mathbf{L}[a, a'] \qquad \text{for all a and a'.} \qquad (3)$$

6. Return to step 2 and repeat the process with this new Concern Vector, **c**, until concern levels all reach zero or are below a threshold value, v.

$$\text{Repeat until } \mathbf{c}[a] \le v \qquad \text{for all } a. \tag{4}$$

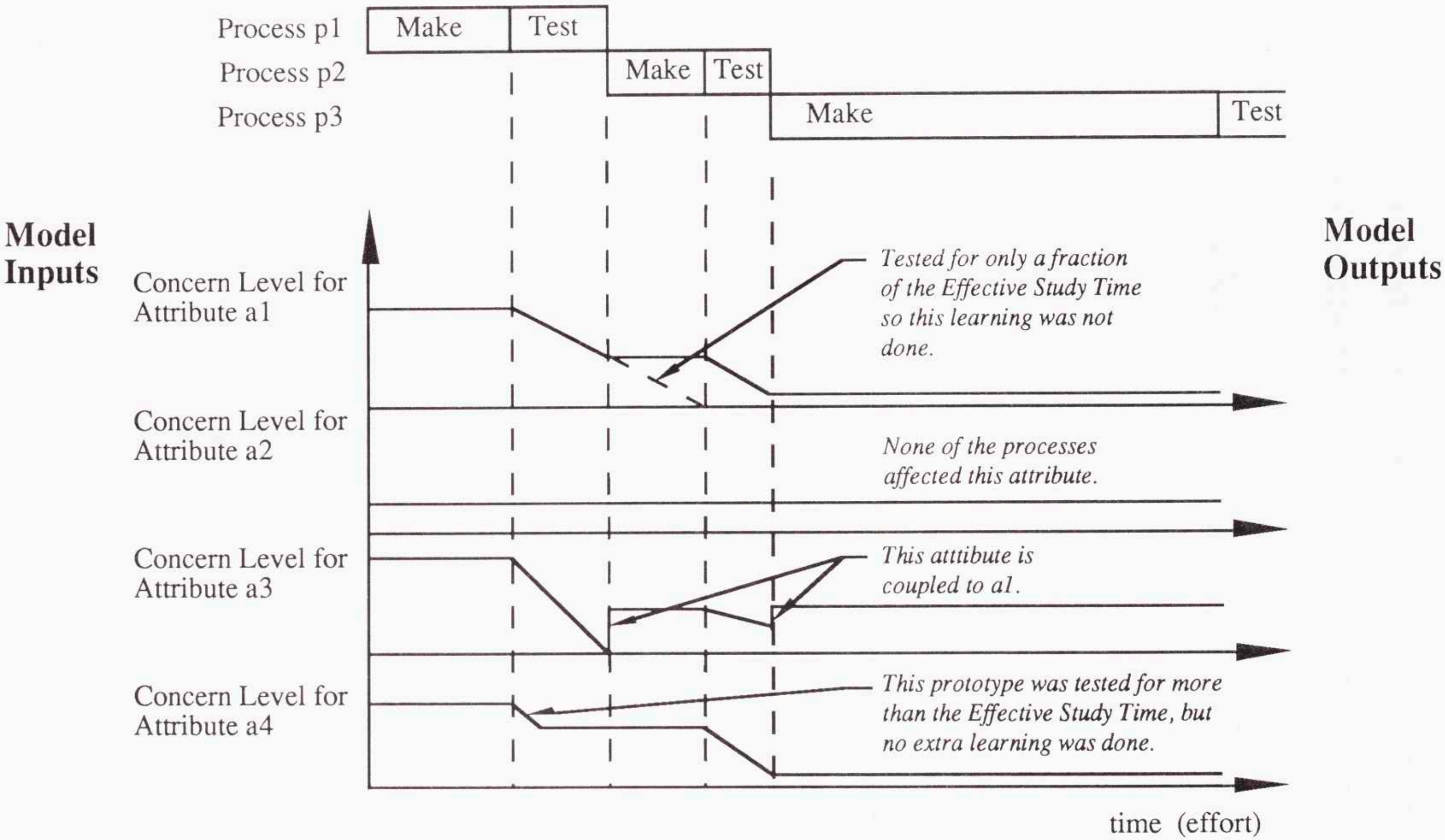


Figure 13. Graphical Example of Using the Model

COMPUTER MODEL SIMULATIONS: THE SEARCH FOR THE BEST PROTOTYPING SEQUENCE

To validate the model with my research and expectations, I wrote a c program to perform the calculations. Appendix D includes the program code listing. The program consists of two parts, an engine and a shell. The engine performs the calculations described above.[2] The shell keeps track of the concern levels, development time, and other record keeping. Simple revisions to the shell will allow the same engine routine to be used to find the best possible prototyping sequence, evaluate a particular sequence of prototype processes, or test heuristics.

The shell in the program listing in Appendix D finds the five best prototype sequences for each possible starting vector of concern levels. The code uses four concerns attributes and five concern levels, this gives 625 (= 5^4) base cases. (I also experimented with ten attributes and 3 levels.) Best is defined as the sequence that reduces the concern levels below the threshold with the shortest development time. The shell generates a starting concern vector and then generates and tests prototype sequences with a recursive routine and the engine. The routine searches all possible combinations of prototyping processes. The search would be exhaustive except that searches further down any branch of the recursive sequence generation are stopped if that branch already has a longer development time than the current best sequence. Figure 14 is a flow diagram of the program.

---

[2] The engine is also easily coded in a spreadsheet. This was done. The spreadsheet version was good for evaluating a particular sequence of prototype processes and generating graphs of the time sequence. But, it was too cumbersome to use for any other model validating.

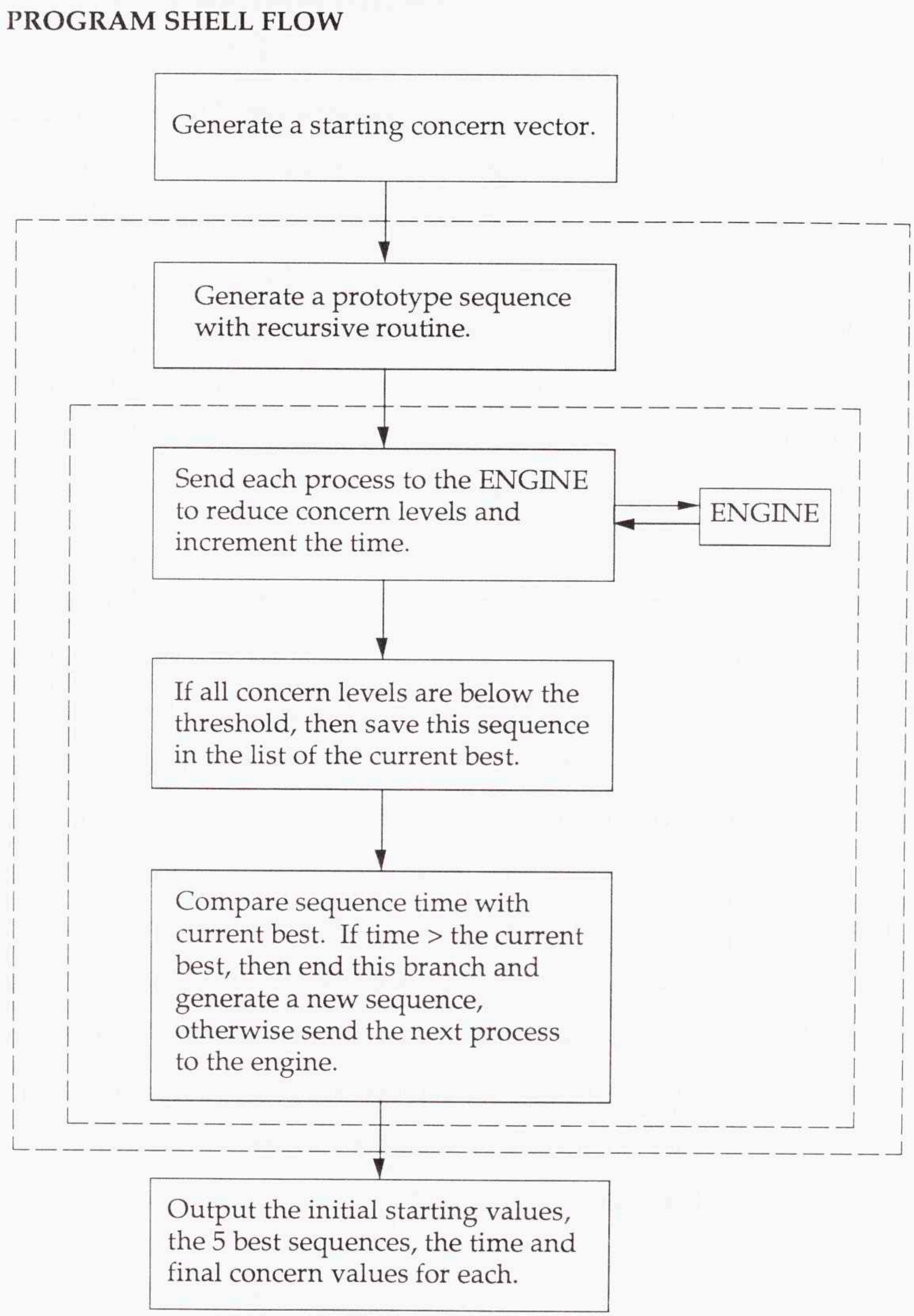


Figure 14. Flow Diagram of Computer Model Simulation (left half)

ENGINE FLOW

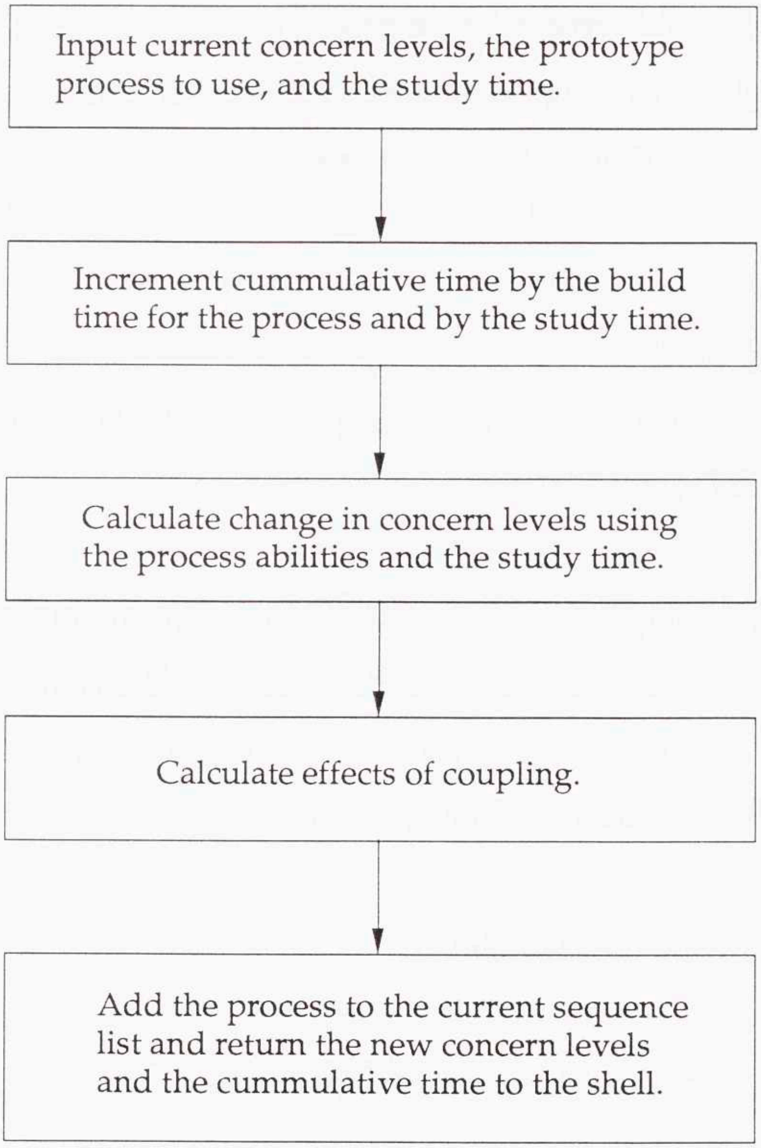


Figure 14. Flow Diagram of Computer Model Simulation (right half)

The starting values and process characteristic values are easily changed in the data structures section. Some assumptions that were made for this program include the following:

- The performance characteristics of the prototype processes used in the computer model are all rough estimates. The level of approximation of the estimates should not effect the insights gained from exercising the model.

- The actual study time, t_s, for each prototype was a defined fraction of the effective study time. In the code listing of Appendix D, $t_s = 0.5$. I also experimented with $t_s = 0.75$.

- The concern levels only had to get below a defined threshold, v, for the sequence to be complete. Because the model reduces the concern levels by percentages, zero will never be reached. In the code listing of Appendix D, v = 0.1. I also experimented with v = 0..25.

- Coupling effects were not included.

- The number of processes in a sequence and the search depth were limited by the computer's operating capacity. The search space for each starting concern level base case had an upper limit that was a function of the number of processes available. In the worst case, the recursive search routine examines the entire space. In the code listing of Appendix D, the number of processes was eight. This gives a maximum search space of 16,777,216 ($= 8^8$) sequences for each base case.

Even though the search was bounded by checking sequences against the current best, the program frequently crashed the workstation it was running on. I experimented with limiting the search space by constraining the number of times a process can be repeated, but this constraint seemed to artificial to include in the model.

RESULTS OF COMPUTER MODEL SIMULATIONS

Appendix D also include the listings of portions of the computer model output from runs using various values for some of the parameters.

In the computer output, the optimal sequences had certain processes repeating a high number of times. The repeating processes are ones that have short build times and short effective study times. Further analysis of the model sheds insight on this result.

When a process is repeated, each iteration decreases the concern level by a percentage. Then equation (2b), the formula for concern level decrease is similar to a declining loan balance at a given interest rate:

$$c[a] = c[a]_0 \bullet (1 - \mathbf{A}[p, a])^n \qquad \text{for all a,} \qquad (5)$$

where

n = number of times the process, p, is repeated.

A graph of equation (5) is shown as the repeating process, p", in Figure 15. A nice property of this formula is that it includes the diminishing returns that results from repeating a process.

Figure 15 shows the key trade-off in the process sequence evaluation of the computer model. The trade-off is between using a single process, p', and repeating some other process, p". Using a single process, p', is represented by the linear decrease in concern level, which is described in equation (2a). Repeating the process, p", is represented by the decaying exponential, which is described in equation (5). The point (c^*, t^*) represents the critical point between the two choices. If the threshold, v, is greater than c^*, then the repeated process, p", is used. If the threshold, v, is less than c^*, then the single process, p, is used. Below is a summary of that point:

$v > c^*$	Use the repeated process, p".
$v < c^*$	Use the single process, p'.

This creates a "strict dominance" where some processes can be eliminated entirely from the process list because they will never be in the optimal solution. As more attributes are added, the analysis is multi-dimensional but essentially the same.

The problem with this result (all problems can be solved by repeating a single rapid process) is that it does not match a designer's expectations. This issue arises because the model recognizes a process's ability to solve some percentage of the problems in a category. But, being able to solve a percentage of concerns is not supposed to mean that it can solve every type of concern in that category; it means that it can solve some types of problems in that category. Here is an analogy of the problem: If I told the model that my car runs 10% better if I change the tires, then the model would tell me that my car runs 95% better if I change the tires seven times ($1 \bullet 1.1^7 = 1.95$).

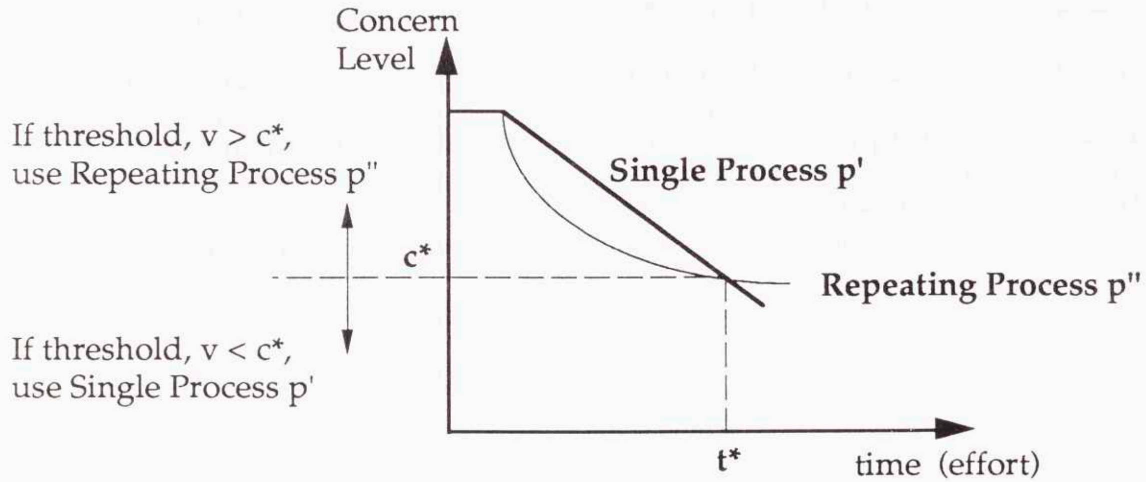


Figure 15. Choosing between a Single Process and a Repeating Process
Use the process that gets below the threshold, v, in the least time. The critical point between the two processes is (t*, c*).

One way to address this issue is to redefine the concern categories so that each category contains only attributes that can be addressed by a certain process. This solution is not acceptable because it forces a process to be used whenever a category contains a concern. Alternatively, one could define categories in such a way that any of the processes can either solve the problem completely or it can not solve it at all. The problem space is too broad for me to make that kind of category definition.

The other result from experimenting with the model was that the coefficients in the Coupling Matrix, L, can easily overpower any effects of using the prototypes. If the coefficients are too large and the matrix allows coupling between too many attributes, it is easy to end with higher concern levels than the beginning ones. It's as if one is hypersensitive to any change. Care must be taken in selecting the coefficients. Values less than one concern unit ("problon") seem to work satisfactorily.

INSIGHTS FROM PROJECT AND MODEL

The results of this work can be divided into these parts:

- How to think about prototyping. This includes a section describing a metaphor for prototyping and one on how the model can contribute.
- How to do prototyping more effectively. This section considers some prototyping heuristics and strategies.
- What are the future needs of prototyping.

These topics are discussed in detail below.

INSURANCE METAPHOR FOR PROTOTYPING

One of the lessons learned during my observations and discussions of prototyping was the metaphor of prototyping as insurance. This is a good way to think about prototyping.[3]

- A person buys insurance to reduce the loss from some unknown threat such as theft or fire. In the same way, a designer can use prototypes to reduce the possibility of expensive changes in the future.

[3] My friend Tim Coonahan inspired the insurance metaphor. He sells prototyping services. Now he says he feels like an insurance salesman.

- The homeowner buys insurance for the events that she perceives to be likely to occur or events that have consequence that she can not afford. The designer makes prototypes to check for possible problems so expensive and time-consuming changes do not have to be made in production tooling.

- The right balance of insurance is important for individual consumers. In economics, the insurance balance is calculated using risk-affinity curves. The proper amount of insurance (the shape of the curve) changes throughout a person's life as career, family, and life goals change. The correct amount of prototyping is also important for the designer. As time-to-market and shorter product life cycles become more common, cost and risk must be balanced to achieve optimal product development schedules.

The question a designer should ask is: How much prototyping and analysis are we going to do? Or to put it another way: How much insurance are we going to buy?

CONTRIBUTION (VALUE) OF THE PROTOTYPING MODEL

Another way to think about prototyping is with the model described above. Uses for the prototyping model include the following:

Insight Model

The original purpose of developing the prototyping model was to gain a better understanding of how prototypes are used and to develop effective strategies for using the many different prototyping technologies available. The model does not claim to "tell the future" for a given part following a particular prototyping strategy. In fact, the number values used for the process abilities and times are order-of-magnitude estimates with large variance bands.

The model is useful for developing an understanding of how a particular sequence of prototypes will check and recheck various part attributes. This is measured with the "Concern Levels". A designer can use the model to help place a value on a particular prototyping process during the product development cycle.

Common Discussion Language

The prototyping model and the Concern Levels give the design team a common language to discuss the product development. For instance, a designer might point out that a particular type of prototype is required because it will help lower a certain set of Concern Levels. This gives the design team a framework for planning the product development strategy. A development team could explicitly track the concern levels to help prioritize their progress and remaining tasks.

Link to CAE

A long range use of the model (and certainly a more ambitious one) is to integrate it with a computer-aided-engineering (CAE) system. One could imagine a designer inputing a set of concern levels for a part, and a computer returning an optimized prototyping strategy and the expected development times. Unfortunately, at present, the model in not refined enough nor is the data characterizing the prototype processes accurate enough to make such a system practical. However, fitting the framework to the processes available to a particular design environment is certainly feasible.

PROTOTYPING HEURISTICS AND STRATEGIES

This is the section that considers heuristics and strategies to do prototyping more effectively. One of the primary goals of this study was to develop heuristics, "rules-of-thumb", that designers could use when prototyping new products. There were two main sources of these strategies:

- Strategies observed in practice
- Strategies the prototyping model suggests

In addition, there are some strategies that suggest themselves as possible, although they are not always practical strategies.

The different strategies fall into three general categories. These categories are also a very concise summary of the recommendations of this thesis:

- Learn fast and keep moving
- Reduce product risk
- Plan ahead

The details of these points are discussed below and summarized in Table 6.

Table 6. Summary of Prototyping Strategies

Learn Fast and Keep Moving
- Do Those Processes that Teach You the Most, the Fastest.
- Always Do Something.
- Decrease the Concern Level by One of Several Algorithms.

Reduce Risk
- Repeatedly Run the Low Cost Prototyping Processes.
- Test as Many Different Attributes as Quickly as Possible.
- Progress to Increasingly Better Representations of the Part.
- Concentrate on the Coupled Attributes.

Plan Ahead
- Consider the Sequence of Prototypes Instead of Simply the Next One.
- Prototype with the Final Production Mold in Mind.

Some Other Ideas for Consideration
- Select Prototype Processes Only to Meet Other Constraints.
- Use Your Favorite Process.

Learn Fast and Keep Moving

Do Those Processes That Teach You the Most, the Fastest

One strategy is to always use the process with best total learning/total time ratio. In the model run, it was the short, powerful processes that dominated the "best" sequences.

The most powerful process, the one with the greatest ability to represent the part is, of course, the actual production process. If the final product is quickly available and easy to fix, one would not bother to prototype. A good example is output from laser printers. Every time a document is printed, it can function as the final product. Because the final product is easy to achieve, there is less reason to prototype a document with a different, cheaper, faster, but less "production-style" printer.[4]

There are many factors that determine the speed or price of a process. Besides the basic process capabilities, one must also consider such factors as the availability, past experience, and appropriateness of the process. For instance, if you own the CNC machine in the next room, then CNC may always be the best prototyping process for you to use. It will always be the cheapest and easiest.

---

[4] One may argue that the display on the computer monitor is a better cheap, fast prototype. This is true. However, my example refers to the common practice of using a printout to carefully proofread a document or as the final version.

Although I did not do it in this work, it may be possible to rank processes in a given situation by their weighted average of the ability to address concerns and speed or some other composite number.

Always Do Something

Although this is only an assumption of the model, it is a useful heuristic to follow. Short product development cycles have become too important in today's world to allow time to pass without some type of progress. A project should always have forward momentum. Unfortunately, sometimes when prototypes are built, they sit on an engineer's desk for days or weeks before being tested. One estimate is that 95% of product development time is waiting in queues [Smith92]. If you're not working on your project, someone somewhere should be doing something for you. There is no time to waste anymore.

Decrease the Concern Level by One of Several Algorithms

There are several other heuristics one could apply to the model. There is no reason to suppose that these are necessarily worthwhile, but I present them to show how the model can be used to generate other heuristics:

- Address as many of the top concerns as possible with each prototype.
- Use a process that addresses the attribute with the highest concern level. Keep testing until it is no longer a concern.
- Use the process that has the steepest gradient to reduce the highest concern level the fastest.

- Address the attribute with the highest concern level until it is less than another concern. Now choose another process to address the new high attribute concern. Repeat.

One observation that applies to all of these strategies is that they specify only the next prototyping process to use. No attempt is made to determine the entire sequence of prototyping during product development. They are very local heuristics.

Reduce Product Risk

Repeatedly Run the Low Cost Prototyping Processes

Take advantage of any prototype processes available that are low-cost or can use underutilized equipment that is considered a sunk-cost. The focus of this idea is to keep checking your design. This becomes more apparent when coupling is added to the model. Every time a change is made, it may affect some other aspect of the part. Even partially representative testing has value because it may find some subtle problem that may otherwise be missed. If the prototyping equipment is underutilized and sunk-cost equipment, there is no cost to use it to test the prototype. An example of this is the CAD interference tests that Boeing performs automatically every night using a very large collection of mainframe computers [Stix91]. Another example is to test the computer-based solids model with all the available linkages (FEA, moldflow, etc.) just because they are available and you might find something. An even simpler and cheaper example is to keep thinking about and sketching the design in your spare time (or during class).

Test as Many Different Attributes as Quickly as Possible

Another class of strategies is to use all the resources available. This "shotgun" approach will reduce risk, because more attributes are tested, however, these approaches must be carefully considered to insure time or other resources are not wasted. Some of the heavy testing strategies include the following:

- Do not let prototyping assets be underutilized. This is the same low-cost argument above. If you keep running the low-cost test, you will constantly retesting attributes for new concerns.

- Use the "hidden" prototypes. The masters for rubber molding and the complex shaped EDM tools are part representations that are sometimes not seen by the designer. These are another opportunity to check the design.

- Show the prototype to EVERYONE. This strategy builds off the team concept. A prototype provides a common discussion language [Leonard-Barton91] so everyone can participate in problem-solving, contribute their expertise, and test for their area of responsibility.

- Prototype several variations of the part at the same time. Examples of this strategy includes trying several wire diameters to size a spring and changing part parameters to perform arrayed (Taguchi) experiments.

- Keep prototyping non-critical path items, especially for interaction with the critical path. This is another opportunity to look for coupling problems with the part.

A very valid concern with these types of heavy testing strategies is to avoid "Hardware Swamp" [Clausing91]. When too many prototypes are around, there may be confusion about which is the correct version. Also, valuable people resources may be too limited to work with all the prototypes.

Progress to Increasingly Better Representations of the Part

A very common sequence observed in practice is one where each successive prototype process is a more complete representation than the previous. For example, a common sequence is:

(1) CAD, (2) Machined Part, (3) Rubber Mold, (4) Production Mold.

There are not many instances were someone makes a machined part once they have rubber molded parts. Why bother when the rubber molded part represents all the attributes that the machined part does and more?

In the above sequence, each process is a richer representation of the final product. Each process tests all the attributes that the previous process tested as well as some new ones.

This strategy was not observed in the model runs, but if coupling is considered, I believe it is a logical sequence. The coupling between attributes can create new concerns. A subsequent and richer prototype will retest attributes and find any new problems created by coupling.

Concentrate on the Coupled Attributes

The model presented contained limited provisions to account for the coupling between different attributes of a part. Unfortunately, it is this coupling that is one of the key problems in product development. Coupling creates the need for iterative steps to arrive at the correct solution [Eppinger90]. Strategies that take coupling into account include the following:

- Decouple as many problems as possible and solve them in parallel. Presumably this will take more resources, but the calender time for the development will be less. In addition, one must beware of unexpected interactions between the alleged uncoupled parts.

- Carefully chose which attributes to couple in the design. DFM rules and concurrent teams created more coupled attributes in the design. The feedback of the iterations in coupled designs are important for the part design and sometimes for organizational learning [Eppinger90].

- Address attributes that are highly coupled first. A good example of a highly coupled attribute is appearance. It is affected by such things as material selection, surface finish, functional geometry, operating requirements and many others.

Plan Ahead

Consider the Sequence of Prototypes Instead of Simply the Next One

It is important to include in the plan of the product development process the prototypes that will be built. The plan, of course, must have provisions for problem corrections and reflection at the project's end so the process can be improve for the next one. One must be sure that all the concerns are addressed in the most effective way so that the final break-even time of the project is minimized. The cost of the product development effort and the prototyping must be included in the break-even time analysis.

When repeated fast processes are better than any one process for a given set of attributes, repeat the fast processes. For example, in some cases, repeated runs of Stereolithography and Mold Flow Analysis can teach you as much as a rubber molded part in less time. Even though there are diminishing returns on the learning from repeated prototyping process, there are situations where repeating a process allows you to learn more and learn it faster. It will be better to repeat than to move on to the next prototyping process.

This heuristic is often seen in computer analysis type processes such as CAD, FEA, and Mold Flow Analysis. Once a problem is fixed, the new design is quickly and easily rechecked. However, the heuristic happens less often in the physical prototyping processes such as Stereolithography or CNC Machining. There seems to be a feeling that once the prototype has been made and repaired as much as possible, it is time to move on to the next process. I argue that there are some situations where much is gained by repeating a process to check the new changes created by the learning of the previous prototype.

Prototype with the Final Production Mold in Mind

The ultimate goal of the prototyping sequence is to reduce the risk, cost, and time of the part development by exchanging cheap testing now for expensive mold changes later. This requires understanding production molds and their problems to address them with the prototype. Keep the final process in mind when prototyping to prevent missing some subtly. For instance, you can't test part interference with a prototype that does not include the mold draft. Some strategies that focus on when and how the production mold is made are included below:

- Prototype only those aspects that are hard to fix in the mold later. In some ways this is analogous to the testing strategy that requires checks before any high-value operation. One complication is that almost everything except perhaps surface finish and very minor dimension changes (in the proper direction) are hard to fix on a mold.

- An aggressive strategy is to start mold making before all testing is done with the expectation that it will be changed later. This strategy could be very effective if the mold making area was optimized to make mold repairs. There are examples where the early mold-start strategy is used. But I do not believe there are many shops that try to optimize their mold-repair processes.

- A very conservative strategy is to start production tooling after prototyping has solved all the possible problems. But, one must determine the level of confidence required to "solve all the possible problems". This level of confidence is the threshold that the concern levels must reach.

In my observations, prototype testing continued even after the mold was ordered. But the timing to order the mold seemed to based more on department practice than on concern levels. I believe there are opportunities to reduce development time by basing the mold order on the concern levels of the parts. For instance, mold frames can be ordered as soon as the general part size is known. The mold making can be paced by the level of concern in the different attributes.

Some Other Ideas for Consideration

These are some heuristics that I observed in practice. I include them because they are frequently used and deserve comment.

Select Prototype Processes Only to Meet Other Constraints

A minimalist strategy is to let other product development characteristics determine the prototypes to be made. The assumption here is that enough prototypes will be made to learn about the part, and there is no reason to do more. Examples of this type of prototyping sequence strategy include the following:

- Make only the prototypes that you are constrained to make for reasons other than testing. Here are some examples:
  - CAD files must be made to record the design.
  - Trade shows require physical hardware to show the customer.
  - Company policy requires a prototype at certain parts of the new product generation cycle.

- Prototype before any high-value design decision is made. This is analogous to the testing strategy of inspecting before any high-value operations are done. There is no sense in performing an expensive operation on a priori bad part. One difficulty is determining the value threshold that requires testing.

- Prototype one part feature and leave room in the prototype to add other features in the future. If features can be decoupled, they can be added to the prototype part one at a time. This serial approach can efficiently test a complex part because each feature is added and tested individually.

Use Your Favorite Process

If one does not consider a prototyping strategy, they will use what is easily available to them. Two examples of this concept are the following:

- As one gains experience with a process, they are more likely to use it again to test new product designs. If one is comfortable with a particular prototyping technology and has accumulated knowledge about it, there is usually some value in using it. There will be no learning time required to use the process. However, it is unlikely that the same process is appropriate to test every part that is designed.

- As new prototyping technologies are developed, they gain an aura of newness and high-technology that makes one want to use them. Stereolithography enjoyed this aura over the last few years, but now its aura is fading as other technologies, similar but newer, are developed. The prototypes from these technologies, by the way, are inexpensive enough that some engineers are ordering prototypes without permission of their supervisor [Machlis91].

FUTURE PROTOTYPING NEEDS IDENTIFIED

Both my observations and the model agree that prototyping can be improved by getting parts that are closer to production value and getting them faster. In my observations, most of the serious problems, those that took the most time to fix, occurred during stress testing. The sooner a part can be functionally tested, the sooner the problems can be solved. Everyone would like to see a laser printer that makes functional parts or, even better, the replicators from the starship Enterprise [Sternbach91]. And, many people are working on this problem [Machlis91, Brown91].

But a better prototyping process does not necessarily have to be one of the stereolithography rapid prototyping type technologies. A faster method to make molds would be equally valuable and perhaps more feasible for production-type parts. Technologies that may play a role here are thermal spray metal [Weiss90], high temperature epoxies, low-melting point alloys, and nickel vapor deposition [Mirotech91]. All of these would require a master that perhaps could be made with one of the traditional rapid prototyping processes. These secondary operations to make functional prototypes quickly could be better than ones that make non-functional parts directly.

Another advantage of making near-production parts is that these parts may be good enough to sell, which will put the product on the market sooner. The extra expense of such parts would be worthwhile if the time-to-market was critical. The first few months of sales might be limited until the high volume processes were running, but at least the product would be on the market. There are some companies using proprietary processes to sell quick-turnaround, functional plastic prototypes for lots of approximately 100 pieces [QMS91, ProDesign91].

Another way to get prototypes faster is to streamline and optimize the area that makes the prototypes. The improvements and any added capacity can be justified by the value prototypes have in reducing product development time.

To summarize, richer representations that are available faster would be very valuable. The attribute to concentrate on is making a part that can function in stress testing.

SUMMARY OF INSIGHTS

This last chapter was rather long so I want to summarize the major points. The observations of practice and exercising the model yielded the following insights:

- ***How to think about prototyping***

  One can use the insurance metaphor that compares building a product to owning something of value. Prototyping is like insurance that protects you against possible future losses.

  The model helps you think about prototyping in the following ways:

  - to gain insight into different prototyping strategies
  - as a common framework of discussion for a design group

- ***How to do prototyping more effectively***

  Many prototyping heuristics and strategies were presented and evaluated. They fall into these categories:

  - Learn fast and keep moving
  - Reduce product risk
  - Plan ahead

- ***What are the future needs of prototyping***

  Valuable prototyping process will be able to make richer representations faster. An area ripe for improvements is technologies that use a non-functional rapid prototypes as a master to make a low-cost, quickly-made mold. Gains can also be realized by optimizing the prototyping production area.

CONCLUSION

This project used observations of prototyping practice to develop a model of prototyping strategy. The model and the observations were used to generate and evaluate prototyping strategies and heuristics.

The metaphor presented for prototyping was an insurance purchase. Prototyping is a cost which decreases the possibility of future higher costs. This is similar to the way insurance reduces the damage of possible accidents in the future.

THE MODEL

A part to be produced is represented as a set of concerns, each of different importance. Product Development is the process of reducing those concerns until the customer is satisfied. Prototypes, broadly defined, are the tools used to reduce those concerns.

A prototype is limited by the capabilities of the process used to make it. A prototype suffers from diminishing returns in utility for learning if it is used for too long.

The improvements made during product development can cause other problems through concern attribute coupling. The coupling is modeled as deterministic, and it can only raise concerns. A stochastic element in the coupling model would be more realistic but less conservative.

The model is useful for the following:

- Insight into prototyping.
- Common discussion language for tracking concerns.

A computer code version of the model was written and used to search for the sequence of prototype processes which most effectively reduced a given set of starting concern levels. The code can easily be modified to evaluate a particular sequence or evaluate heuristics.

The data in the model was for plastic production parts because the tooling is expensive and takes a long time to make. Prototyping is very valuable for those types of production processes. However, to validate the model, one recommendation is to use it for a shorter lead time production process such as sheet metal manufacturing.

INSIGHTS

Although the model uses many assumptions, the insight from creating and experimenting with it combined with the observations of practice provides some valuable insights. The heuristics generated from the work can be summarized with these statements:

- Learn fast and keep moving
- Reduce product risk
- Plan ahead

Anecdotal experience showed the value of having the production group make the prototypes, including those that use processes that are only available outside that group. This gives the production group early exposure to the product. Also, parts of the production tooling can be ordered as soon as the critical part attributes (size, for instance) are finalized.

The field of prototyping technologies keeps changing. And, it will continue to change quickly over the next few years as people try to capture the market for a layering-process rapid prototyping technology. The need is great for prototypes that can be made faster and that are richer representations of the final part. Technologies with high potential are those mold making processes that use a layered rapid prototype as a master to quickly create molds for casting or injection molding. Another method to get rapid prototypes is to design and optimize a production area to provide them. If prototypes are truly valuable to the product development cycle, the extra capacity can be justified against time-to-market cost.

In the end, the model presented here is a framework for product development teams to use for their planning. By fitting their particular needs and available processes into the model, they can gain a deeper understanding of their product development plans. Better understanding leads to more effective work.

Antonsson, Erik K.	1992	"Strategies and Imprecision in Engineering Design," March 19, 1992. Lecture at MIT, Cambridge, MA
Boar, Bernard H.	1984	*Application Prototyping, A Requirements Definition Strategy for the 80s,* New York: John Wiley & Sons, Inc., Preface
Brown, Alan S.	1991	"Rapid Prototyping: Parts without Tools," August 1991. *Aerospace America,* pp. 18-23
Buttrell, Frederick J	1984	*Technology to Payoff,* 1984. Philadelphia: Cresheim Publications Books, Chapters 5, 6, 7 and Postscript
Clausing, Don	1991	"Concurrent Engineering," February 6, 1991. *Design and Productivity International Conference,* Honolulu, Hawaii
Deitz, Daniel	1990	"Stereolithography Automates Prototyping," February 1990. *Mechanical Engineering,* pp. 34-39
Dickerson, K. R. Benton, R. H. Atyeo, M. A.	1988	"Rapid Prototyping Tools and Techniques," October, 1988. *British Telecommunications Technology Journal,* Vol. 6, No. 4, pp. 65-72
Eppinger, Steven D. Whitney, Daniel E. Smith, Robert P. Gebala, David A.	1990	"Organizing the Tasks in Complex Design Projects," October 1990. Working Paper #3083-89-MS, Cambridge, MA: MIT Sloan School of Management

Frischmuth, Daniel S. Allen, Thomas J.	1969	"A Model for the Description and Evaluation of Technical Problem Solving," June 1966. *IEEE Transactions on Engineering Management,* Vol EM-16, No. 2, pp. 58-64
Gardner, Dana	1992	"Design's Watchword for the 90's: Speed," February 24, 1992. *Design News,* pp. 69-72
Griesser, Hans P	1983	"Design and Construction of a High-Speed Squirrel Cage Valve," May 1983. SB Thesis, Cambridge, MA: MIT (well, someone has to reference this)
Ince, D. C. Hekmatpour, S.	1987	"Software Prototyping -- Progress and Prospects," January/February 1987, *Information and Software Technology,* Vol 29, No. 1, pp 8-14
Kawakita, Jiro	1990 (est.)	*The Original KJ Method,* Tokyo, Japan: Kawakita Research Institute
Kutay, Ayday	1990	"Strategic Benefits of Rapid Prototyping Technology," June, 5, 1990. Robotics Institute, Carnegie Mellon,*The National Conference on Rapid Prototyping Proceedings*, pp. 101-110
Leonard, La Verne	1991	"Will Rapid Prototyping be Part of Your Future?" January/February 1991. *Plastics Design Forum,* pp. 15-21
Leonard-Barton, Dorothy	1991	"Inanimate Integrators: A Block of Wood Speaks," Summer 1991. *Design Management Journal,* pp. 61-67
Li, Eldon Y.	1990	"Software Testing in a System Development Process: A Life Cycle Perspective," August 1990. *Journal of Systems Management,* Vol 41, No. 8, pp. 23-31

Little, John D. C.	1992	"Tautologies, Models, and Theories: Can We Find "Laws" of Manufacturing?," February 1992. Working Paper WP#3386-92 MSA, Cambridge, MA: MIT Sloan School of Management
Machlis, Sharon	1991	"Conference: Now's the Time for Rapid Prototyping," December 1, 1991. *Design News,* pp. 40-41.
Maho Machine Tool Corporation	1991 (est.)	"The New Machine for a New Form of Machining," Advertising Literature, Naugatuck, CT: Maho Machine Tool Corporation
Miller, Bernie	1991	"Fast Prototyping Makes Models Sooner and Better," February 1991. *Plastic World,* pp. 44-47
Mirotech Inc.	1991 (est.)	"Moulding a New Way, Nickel Vapour Deposition," Advertising Literature, Weston, Ontario, Canada: Mirotech Inc.
Mogavero, Louis N. Shane, Robert S.	1982	*What Every Engineer Should Know about Technology Transfer and Innovation,* 1982. New York: Marcel Dekker, Inc., Chapter 7
Murotake, David K. Allen, Thomas J.	1991	"Computer Aided Engineering and Project Performance: Managing a Double-Edged Sword," August 1991. Working Paper WP #49-91, Cambridge, MA: MIT Sloan School of Management
Noren, C. S.	1988 (est.)	"Rapid Prototyping Network Management System," Martin Marietta Information and Communications Systems, Information Network Laboratory
Peters, M. H. Williams, W. W.	1984	"Location of Quality Inspection Stations: An Experimental Assessment of Five Normative Rules," 1984. *Decision Sciences* 15, pp. 389-408

Powell, Earl	1989	"Design and Decisions", 1989. *Designing for Product Success,* 2nd edition, Boston, MA: Design Management Institute, pp. 24-25
ProDesign, Inc.	1991 (est.)	"Production Quality for 1-500... Fast and Economically," Advertising Literature, Minneapolis, MN: ProDesign Inc.
QMS Co., Ltd.	1991 (est.)	"An Advanced Manufacturing Process for Prototype Products, QMS Systems" Advertising Literature, Namiki, Kawaguchi City, Japan: QMS Co., Ltd.
Raz, Tzvi	1986	"A Survey of Models for Allocating Inspection Effort in Multistage Production Systems," October 1986. *Journal of Quality Technology,* Vol 18, No. 4, pp. 239-247
Schrage, Michael	1991	"Prototypes: Building Blocks for Entrepreneurs," September 9, 1991. Manager's Journal, *Wall Street Journal*
Smith, Preston G.	1992	"Ways to Quicken your Product Cycle," An interview by Lawrence D. Maloney. February 24, 1992. *Design News,* pp. 74-77
Smith, Robert P. Eppinger, Steven D	1991	"Modeling Design Iteration," April 1991. Working Paper WP #3160-90-MS, Cambridge, MA: MIT Sloan School of Management
Sternbach, Rick Okuda, Michael	1991	*Star Trek: The Next Generation Technical Manual,* 1991. New York: Pocket Books, pp. 91, 109
Stix, Gary	1991	"Plane Geometry," March 1991. *Scientific American,* pp 110-111
Stoll, Henry W.	1988	"Tech Report: Design for Manufacture," January 1988. *Manufacturing Engineering,* pp. 67-73

Sutton, Terry	1992	Conversation with Author after she ran the Boston Marathon (3:08), April 20, 1992.
Teague, Paul E.	1991	"How to Slash Design-Analysis Time," November 18, 1991. *Design News*, pp. 81-83
Trapp, Stephen	1991	"New Concepts in Project Management," July 1991. *Generally Speaking,* Elvira, OH: The General Industries Company, p. 2
Tyre, Marcie J. von Hippel, Eric	1991	"Interaction Finding: A Mechanism for Identifying Problems in Complex Situations," November, 1991, Working Paper 3359-BPS, Cambridge, MA: MIT Sloan School of Management
Ulrich, Karl Sartorius, David Pearson, Scott Jakiela, Mark	1991	"A Framework for Including the Value of Time in Design-for-Manufacturing Decision Making," February 1991. Working Paper WP #3243-91-MSA, Cambridge, MA: MIT Sloan School of Management
Wall, Matthew B.	1991	"Making Sense of Prototyping Technologies for Product Design," June, 1991. LFM Master's Thesis, Cambridge, MA: MIT
Weiss, Lee E. Gursoz, E. Levent Prinz, F. B. Fussell, Paul S. Mahalingam, Swami Patrick, E. P.	1990	"A Rapid Tool Manufacturing System Based on Stereolithography and Thermal Spraying," March, 1990. *Manufacturing Review,* Vol 3, No. 1, pp. 40-48
Wheelwright, Steven C. Sasser, W. Earl, Jr.	1989	"The New Product Development Map," May-June 1989. *Harvard Business Review,* pp. 112-125

PART HISTORIES

A major portion of my internship at Kodak involved visiting designers, engineers, and prototype makers. With their help, I compiled histories on the development of some of their parts. I asked questions like the following:

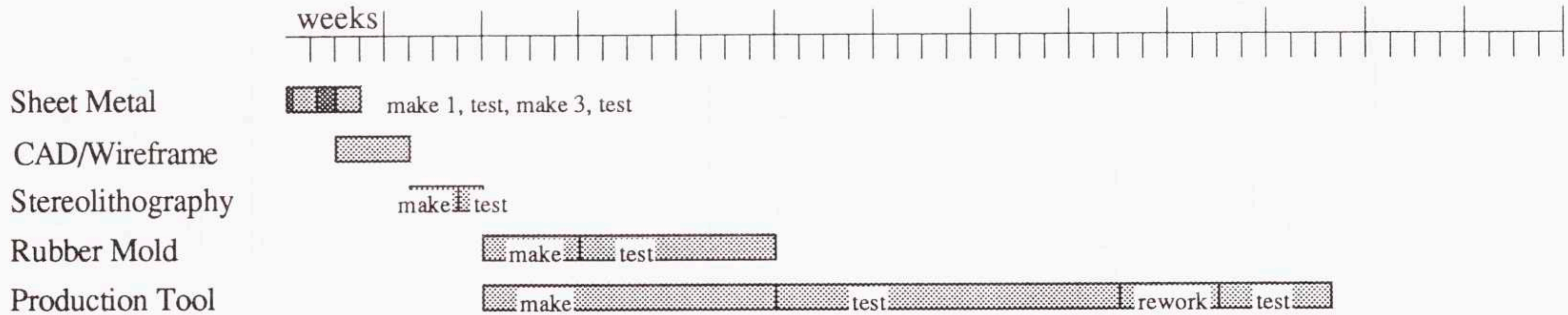
- What part features were you concerned about?
- What prototype processes did you use to develop this part?
- Why did you chose that process?
- What did you learn from the prototype?
- What problems did you encounter?

The information was compiled into the Part Histories presented in this Appendix. Some include pictures of the part. Each Part History contains the following:

- Short description of the part
- GANT chart of the prototype used
- Description of the prototyping strategy
- List of the problems encountered

I also tried to estimate the concern levels for the attributes throughout the development process. My estimates are included in the graphs that accompany some of the part histories. I found this exercise difficult because with hindsight it is hard to objectively determine the concern levels.

TOP HOPPER PLATE



PART HISTORY
Copier Document Feeder paper-handling part.
An air-knife fluffs the paper sheets and a vacuum pulls one sheet down onto this plate. The final part is metal-plated ABS.

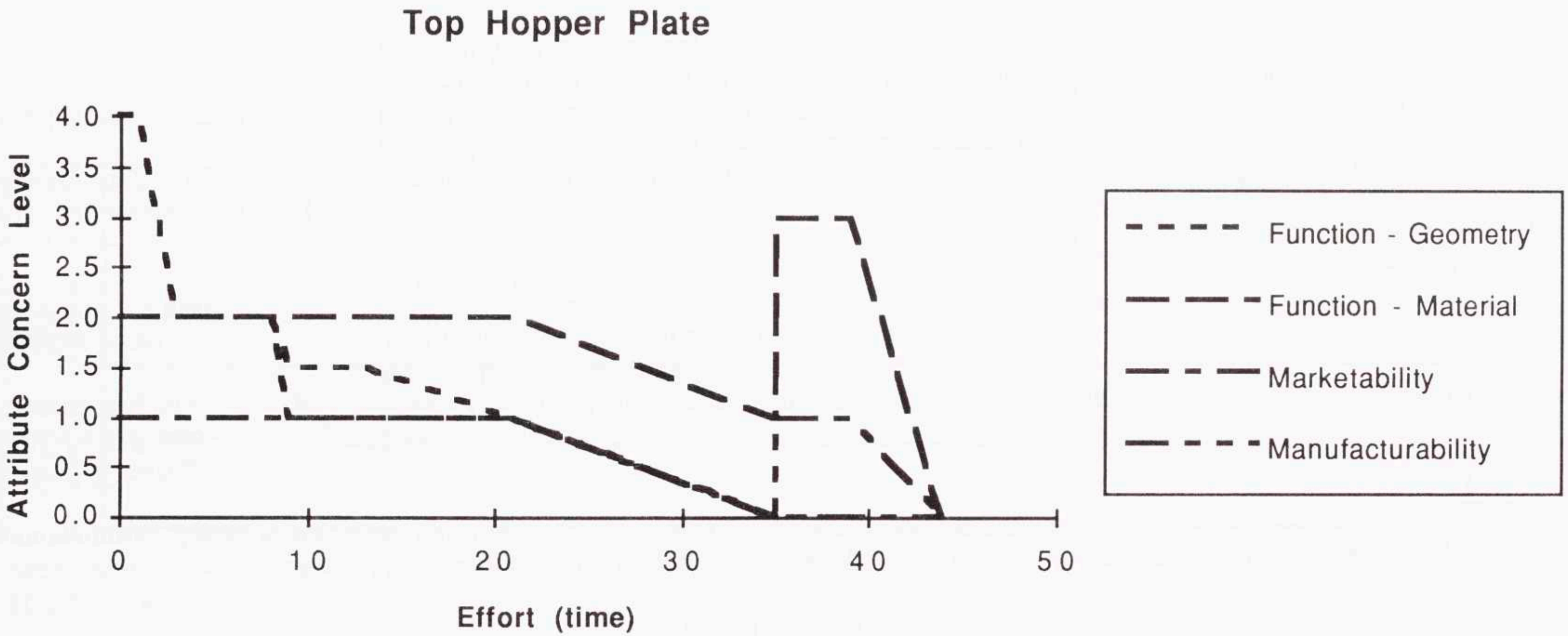
Prototyping Strategy:

Sheet Metal -	Get paper handling angles correct.
Stereolithography -	Rubber molding master and to communicate with moldmakers.
Rubber Mold -	Functional check and parts for system tests by other groups.

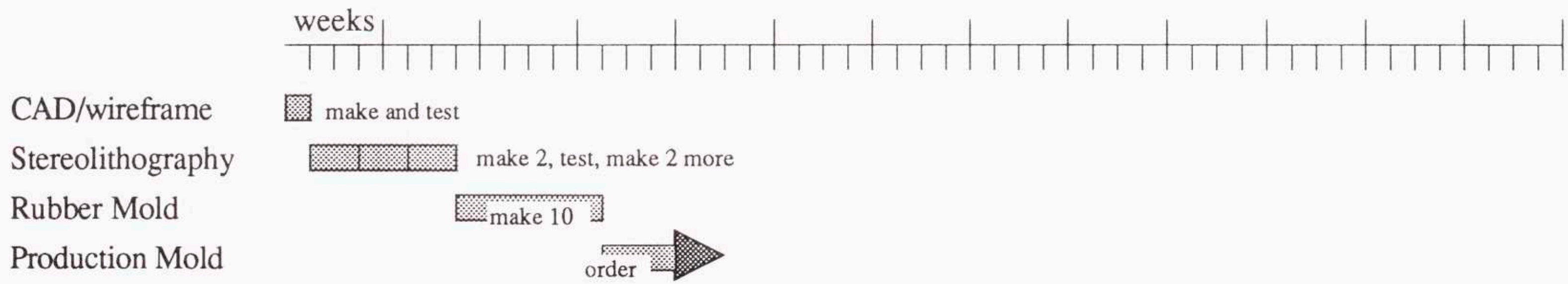
Problems Encountered:

Stereolithography -	Added coring to equalize wall thickness for molding.
	Added slotted holes to ease assembly.
	Added 4 radii in 10 places.
	Learned how to design mating parts.
Production Mold -	Static electricity held paper.
	Wear problems from paper movement.
	Both of these problems were corrected by nickel plating the part.
Nickel Plating -	Assembly problems because of increased pin diameters.
	Decreased friction.
	Plating flash interfered with assembly.
	Plating burrs caused paper jams.
	Still making refinements for cosmetics.

Estimates of Concern Levels:



AIR COLLAR



PART HISTORY

Adjusts vacuum on Copier Document Feeder by turning sleeve to change open slot length. A set screw holds the sleeve in place. There is an unconventional thread to screw the part into the copier wall. This was an existing part redesign.

Prototyping Strategy:

CAD/Wireframe - Design part.

Stereolithography - Check Geometry and experiment with form, CNC machining could not easily make the thin walls. It was also a rubber mold master.

Rubber Mold - Test function in several systems.

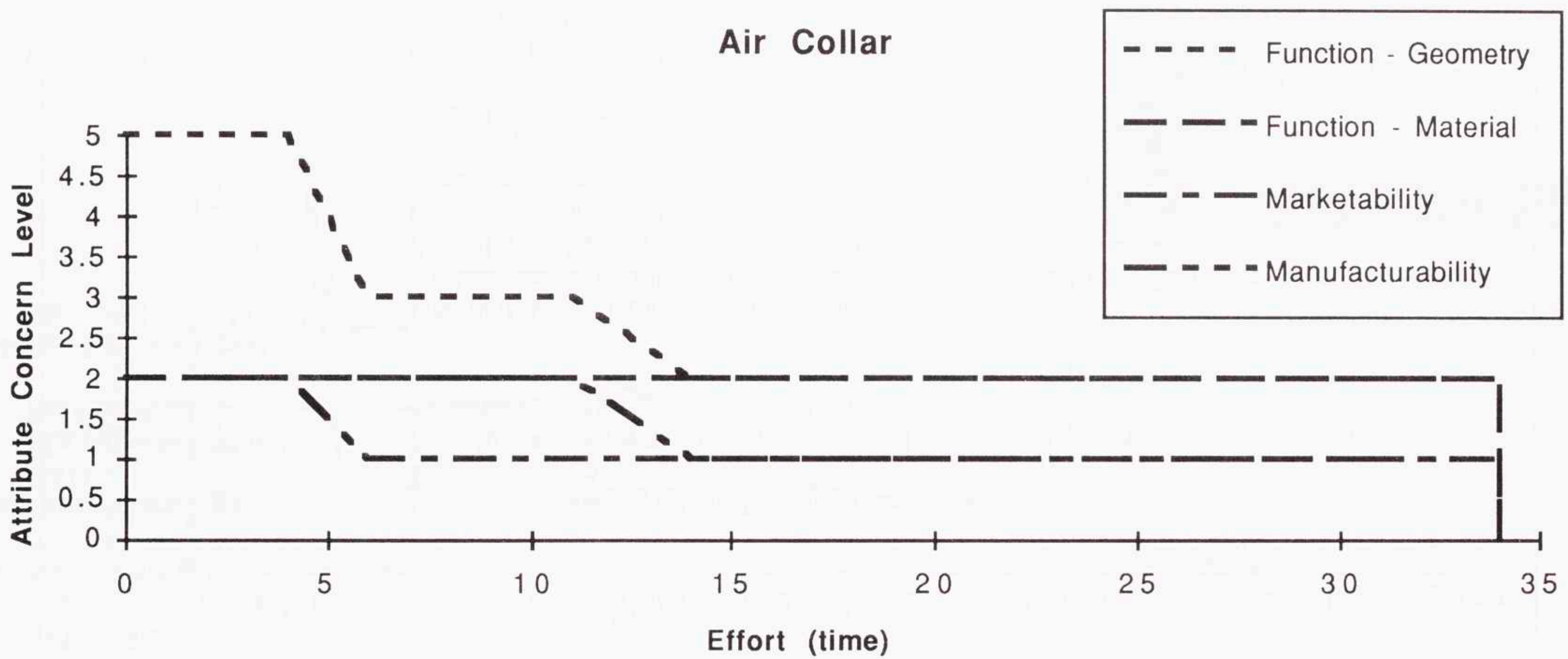
Production Mold -

Problems Encountered:

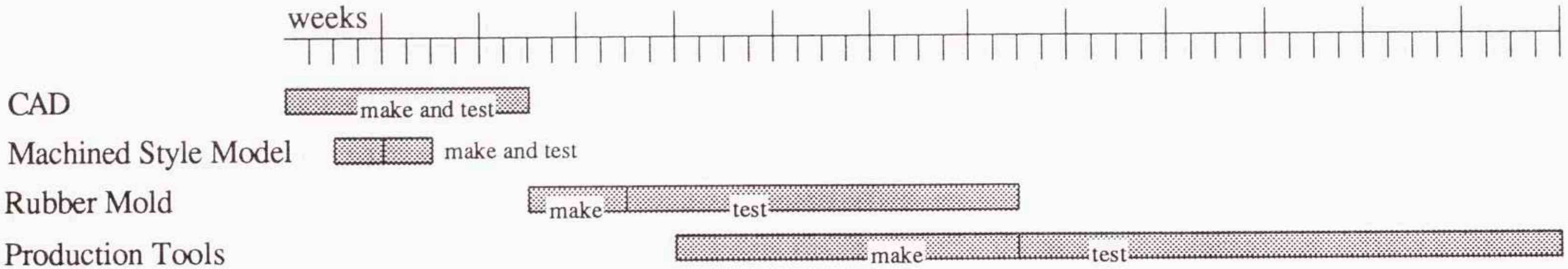
Stereolithography - When adjusting, the sleeve is turned so it is completely closed, the far side of the slot reaches the near side of the mating slot and creates an opening (aliasing).

Rubber Mold - Loose fit in wall created leaks. These were fixed with duct tape.
There is still too much slot adjustment. They are investigating using some other more compliant material for a better seal.

Estimates of Concern Levels:



PORTABLE ID



PART HISTORY

Health Science part. Portable unit that exposes a corner of a medical X-ray plate with the patient's ID. Unit is lightweight and inexpensive. This is a complete assembly, but I included it because it is a fairly complete prototyping story.

Prototyping Strategy: (Engineer's quote: "SLA and FEA might have helped.")

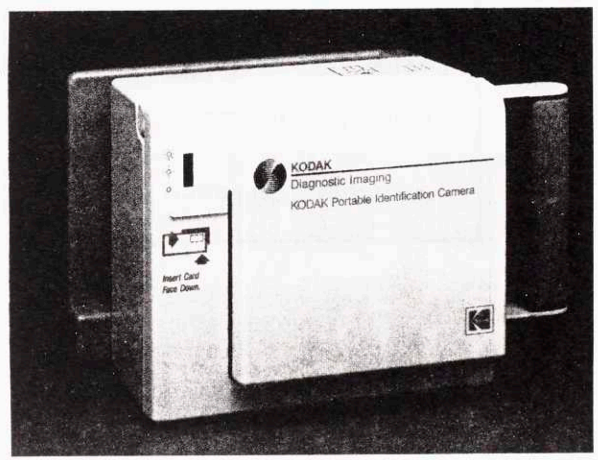
- CAD - Record Design.
- Machined Style Model - Non-functional, machined, painted, and assembled for Marketing approval.
- Rubber Mold - 12 units for engineering tests and Trade Shows.

Problems Encountered:

Machined Style Model -
- Decided where handle should be placed.
- Added core to handle to make constant part wall thickness for better injection molding.
- Added a backup support for lid latch.
- Changed indicator lights to tone.
- Exposure intensity indicator mounted on internal PCB.
- Rotated graphics 90°.
- Added "how-to" graphics.

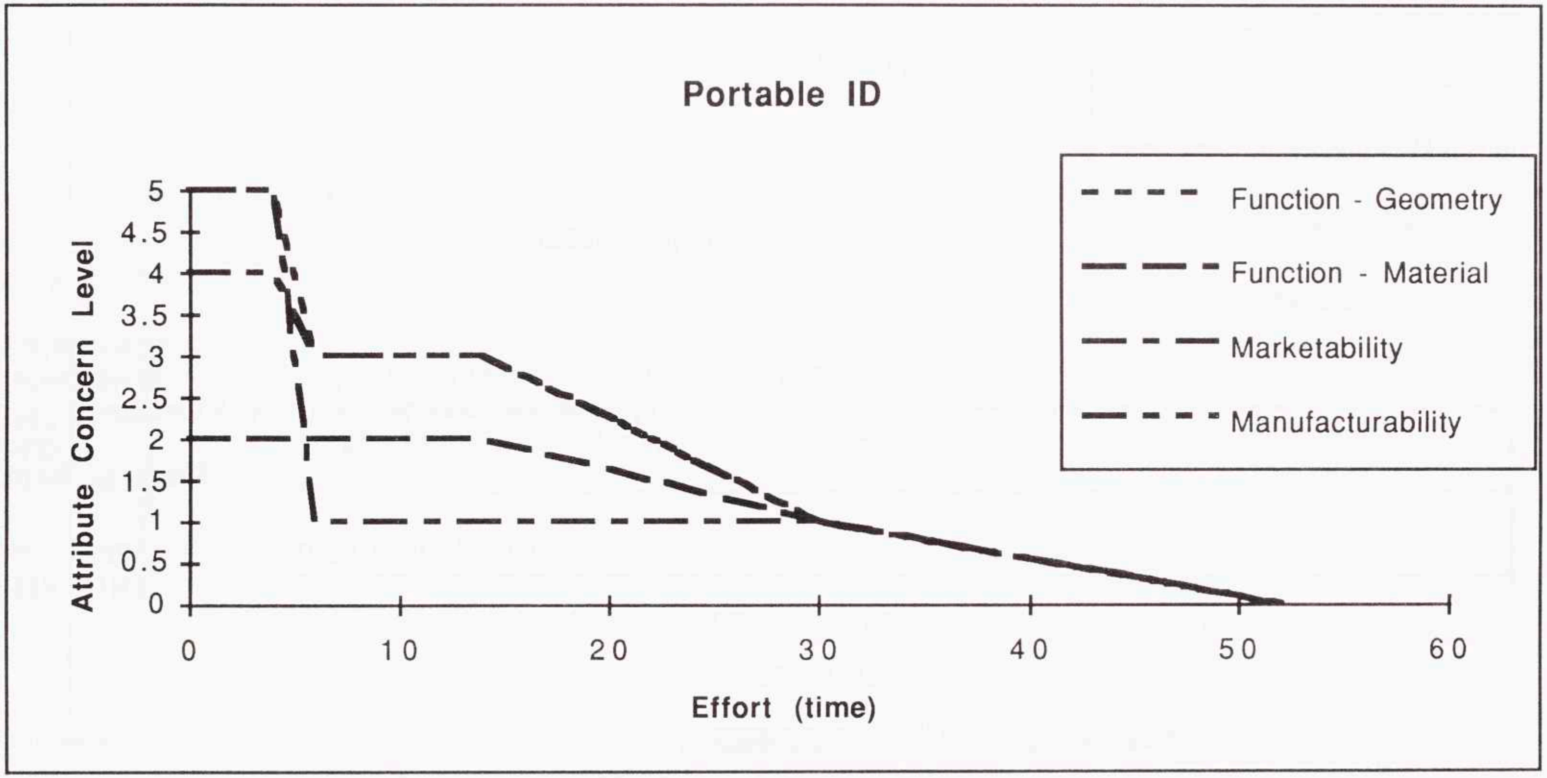
Rubber Mold -
- Changed handle and holster to one integrated part.
- Found light leakage.
- Latch would not close.
- Did not paint entire inside.

Production Mold - Actuator broke in the field, changed sheet metal parts.

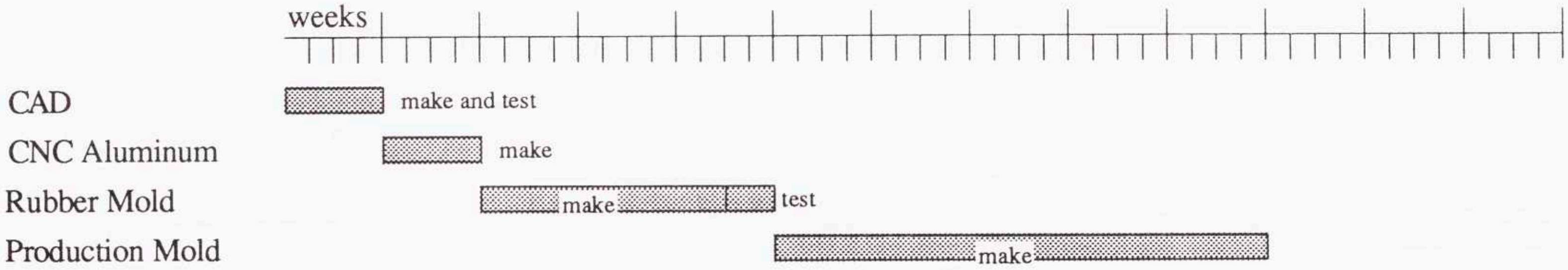


Portable ID Camera

Estimates of Concern Levels:



CASSETTE WIPER



PART HISTORY
Copier Part. Holds roll of "tissue" to wipe copier drum.

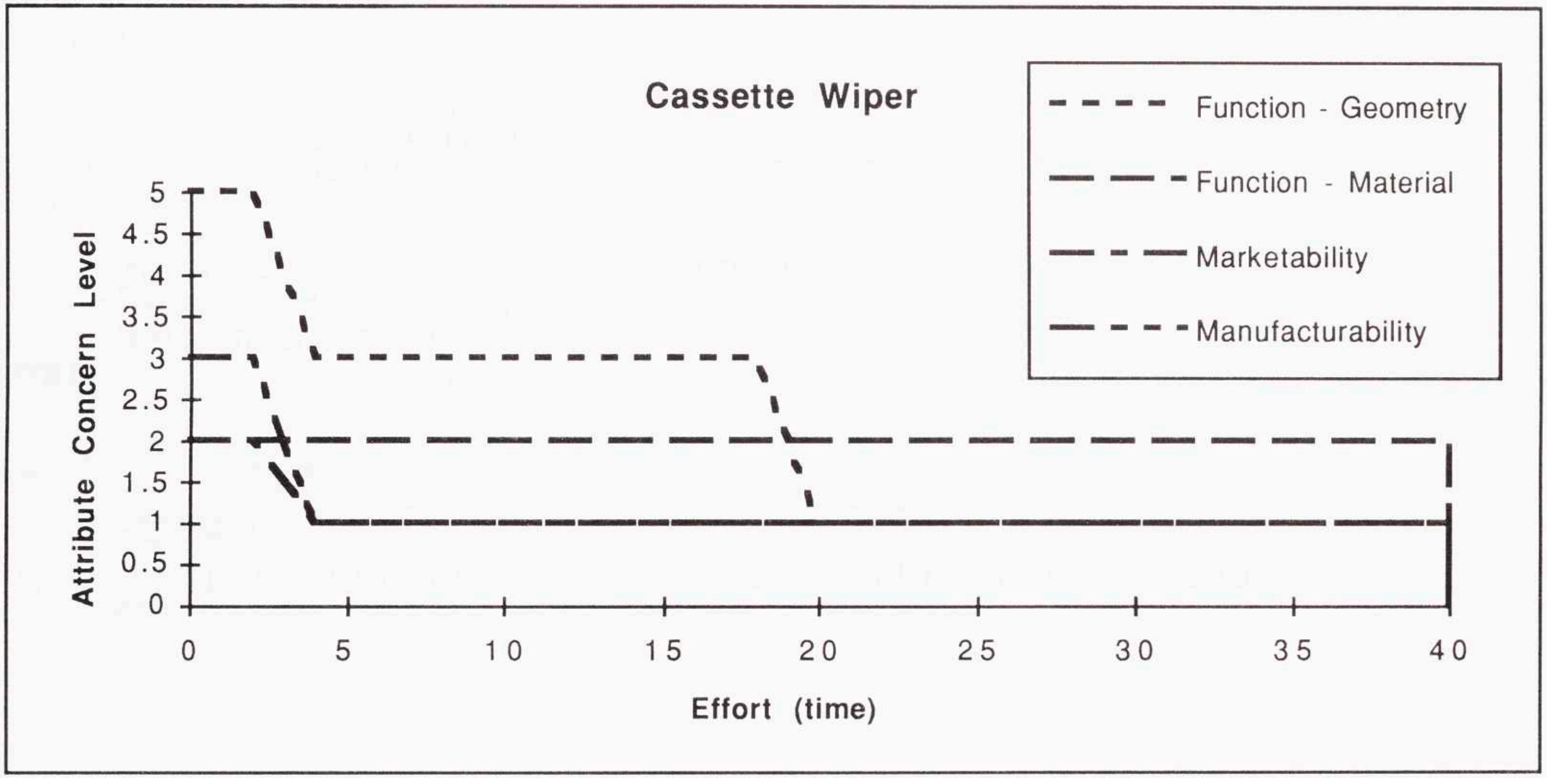
Prototyping Strategy:

CAD -	Design and record.
CNC Aluminum -	Master for Rubber Mold.
Rubber Mold -	20-25 pieces made for system integration.
Production Mold -	

Problems Encountered:

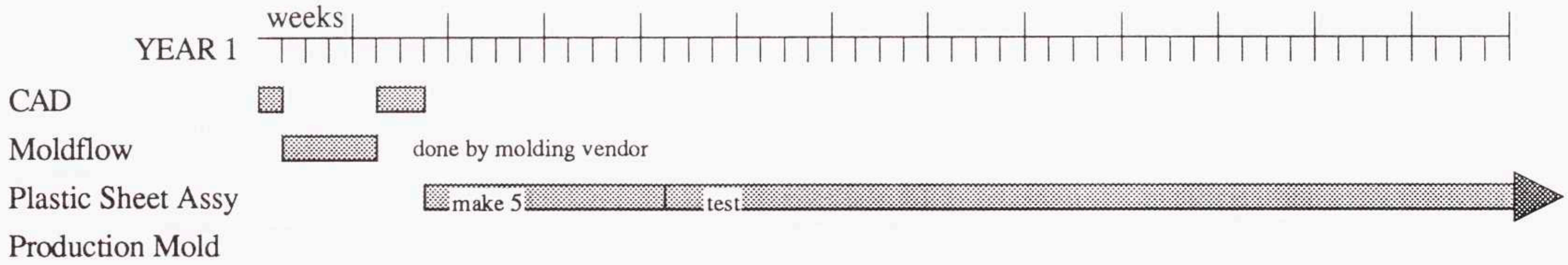
Rubber Mold -	Changes to improve fit in assembly.

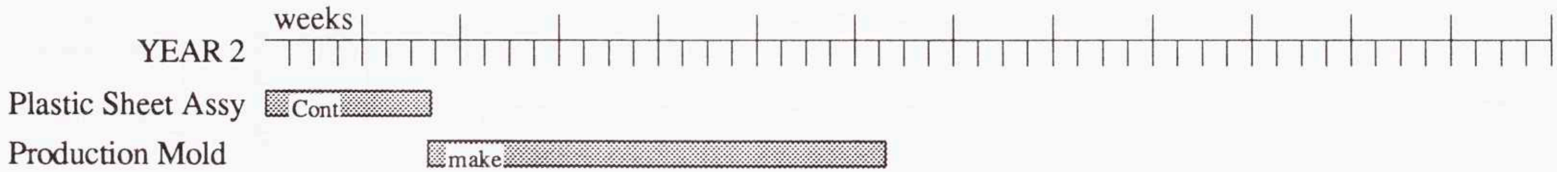
Estimates of Concern Levels:



107

COPIER ENGINE FRAME





PART HISTORY

This is a large part (approx 2' x 2' x 1') that is the structural frame that holds the copier film core.
It was previously made of sheet metal; this is the first molded version.
The design philosophy and primary goal were to prevent production tool changes.

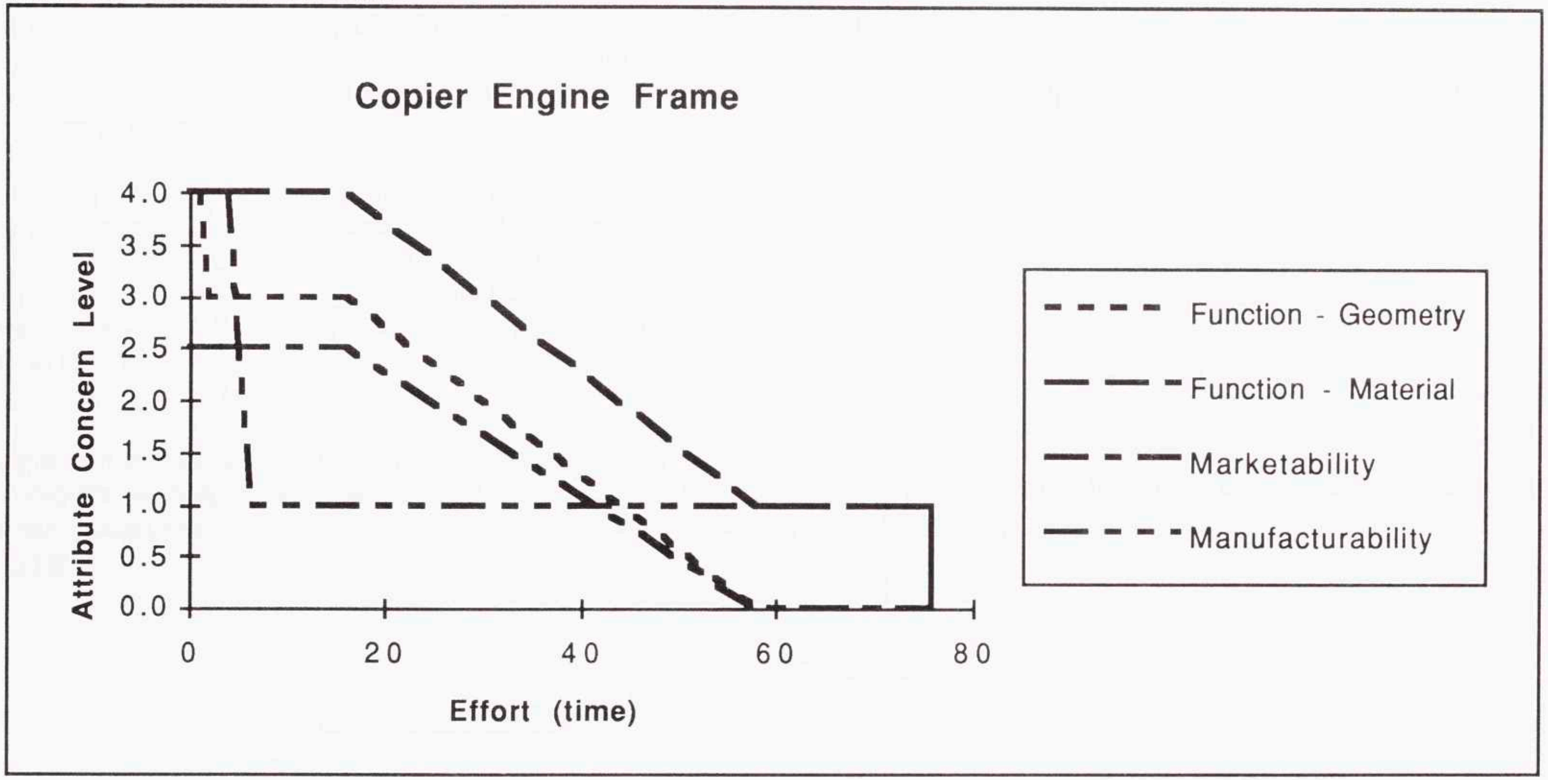
Prototyping Strategy:

Mold Flow Analysis -	Done by vendor because of large part molding concerns.
Plastic Sheet Assy -	Functional Check.
	Show to Manufacturing and other groups.

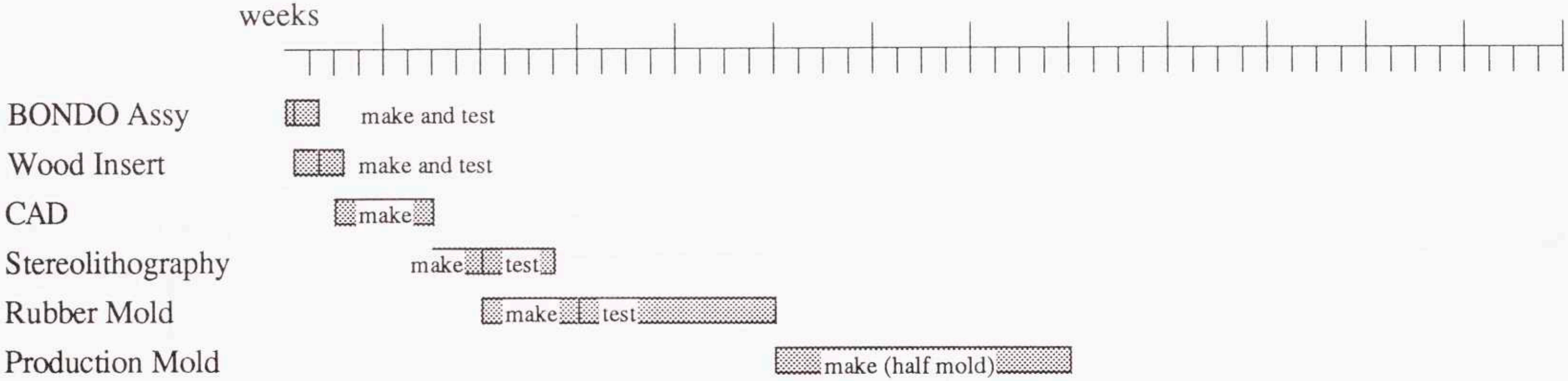
Problems Encountered:

Mold Flow -	No major changes.
Plastic Sheet Assembly -	20 feature locations changed because of surrounding part changes.
Production Mold -	None.

Estimates of Concern Levels:



AIR KNIFE



PART HISTORY

Copier Document Feeder part which blows air to fluff paper and separate the next sheet to be fed.
Draws in the mold are needed to make the holes in the part. This was an alteration of an existing part in an effort to save the tool. In the end, only half the tool needed to be changed (the expensive part).

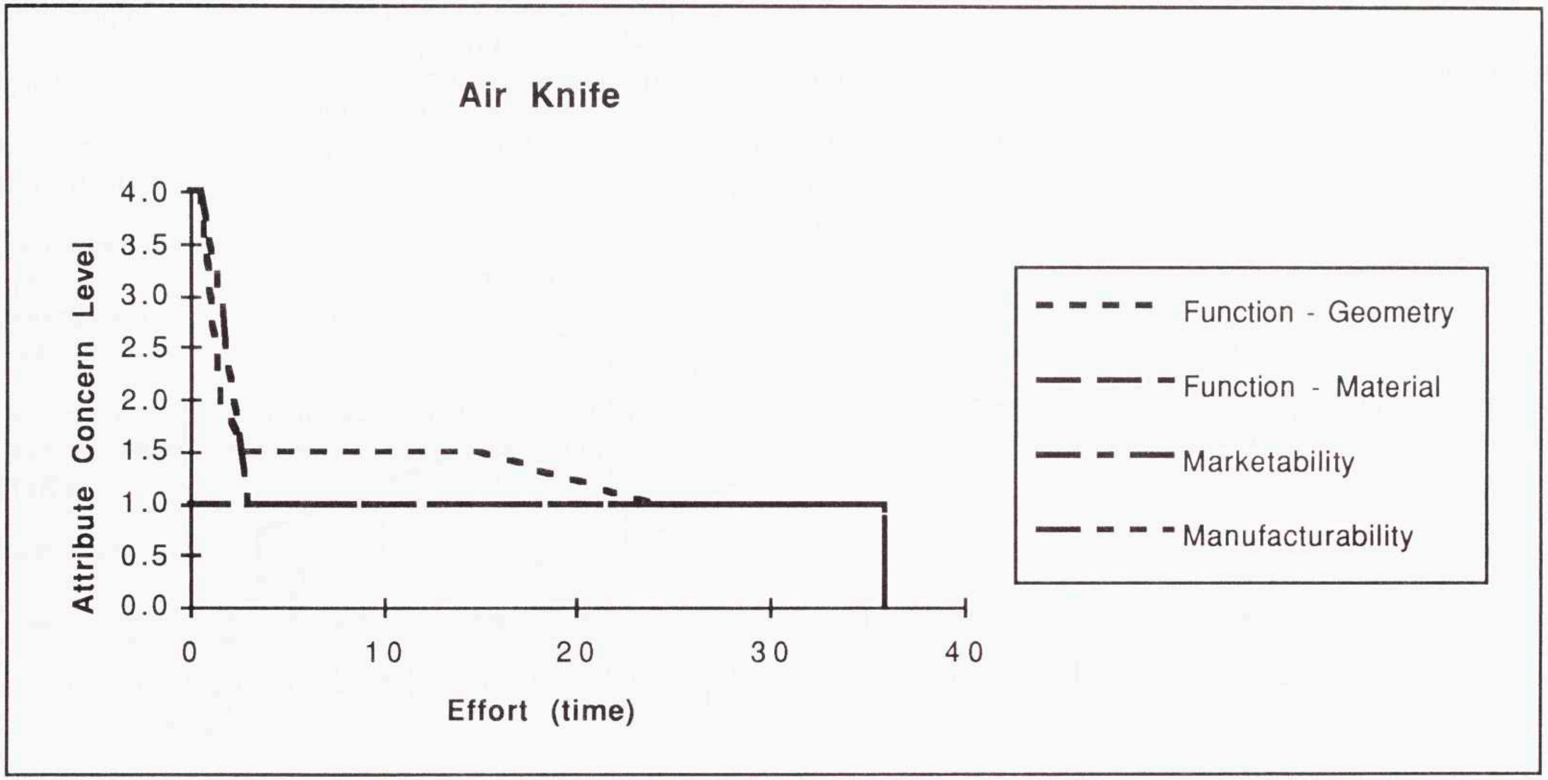
Prototyping Strategy:

BONDO Assy -	Try air holes, diameter and direction.
Wood Insert -	Try accurate drilling of air holes.
CAD-	Record location of air holes.
Stereolithography -	3 pieces, one for the moldmaker and two for rubber molding masters.
Rubber Mold -	Engineering test.

Problems Encountered:

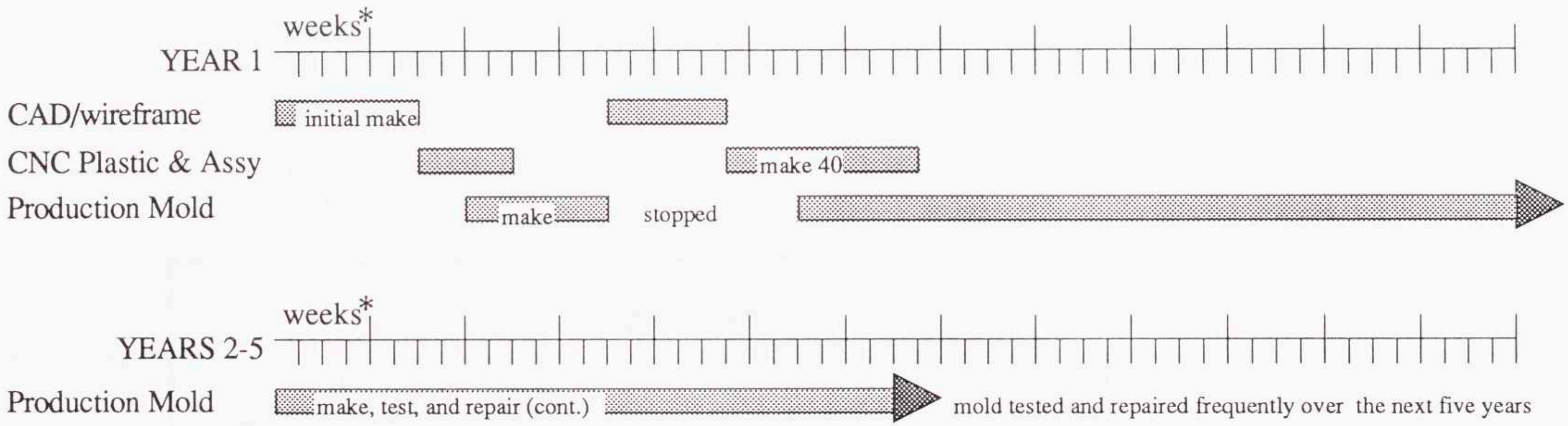
CAD -	Difficult to measure actual hole locations. Instead the holes were fit into the part approximating the wood prototype locations.
Rubber Mold -	Paper handling ribs were refined to keep the paper down. Paper movement surface was refine to be more tolerant of curled paper.
Production Mold -	Tolerance build up problem; mounting on copier frame was changed.

Estimates of Concern Levels:



111

FRONT AND REAR END PLATES



*all times are estimates

PART HISTORY

Copier Parts. Side plates for copier subsystem. Very complex with many features. These parts hold 20 internal components between the two plates and drive gears on their backs. This was critical-path item for the system. Design goal was not to have any side-draws.

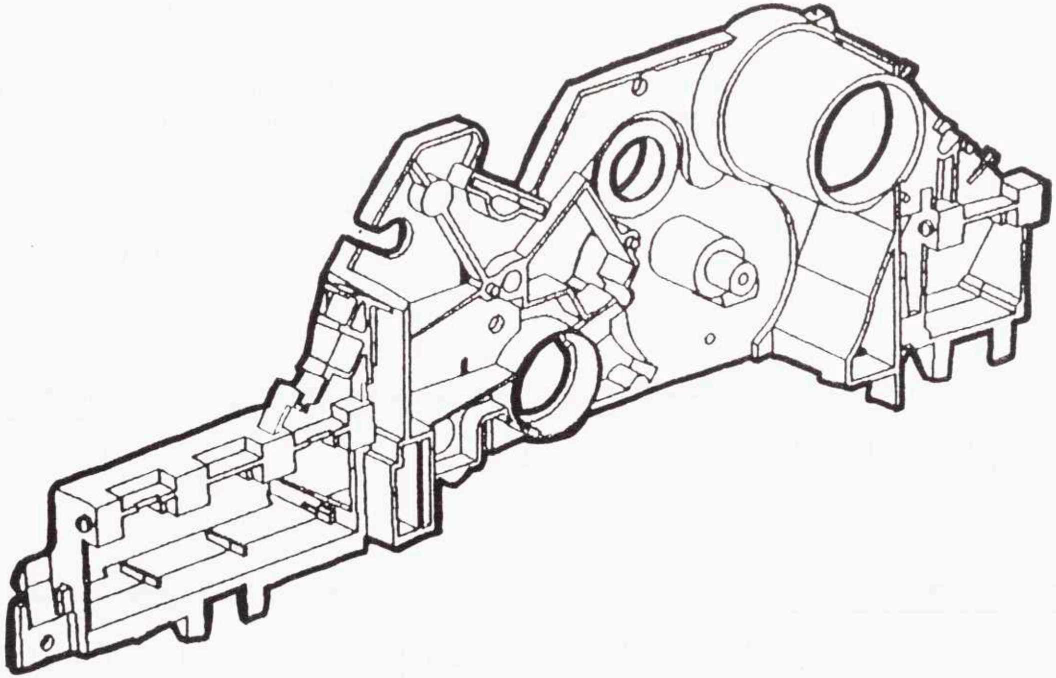
Prototyping Strategy:

CAD/Wireframe -	Design part.
CNC Plastic -	Check parts in assembly, and make parts for system testing and early system prototyping.
Production Mold -	

Problems Encountered:

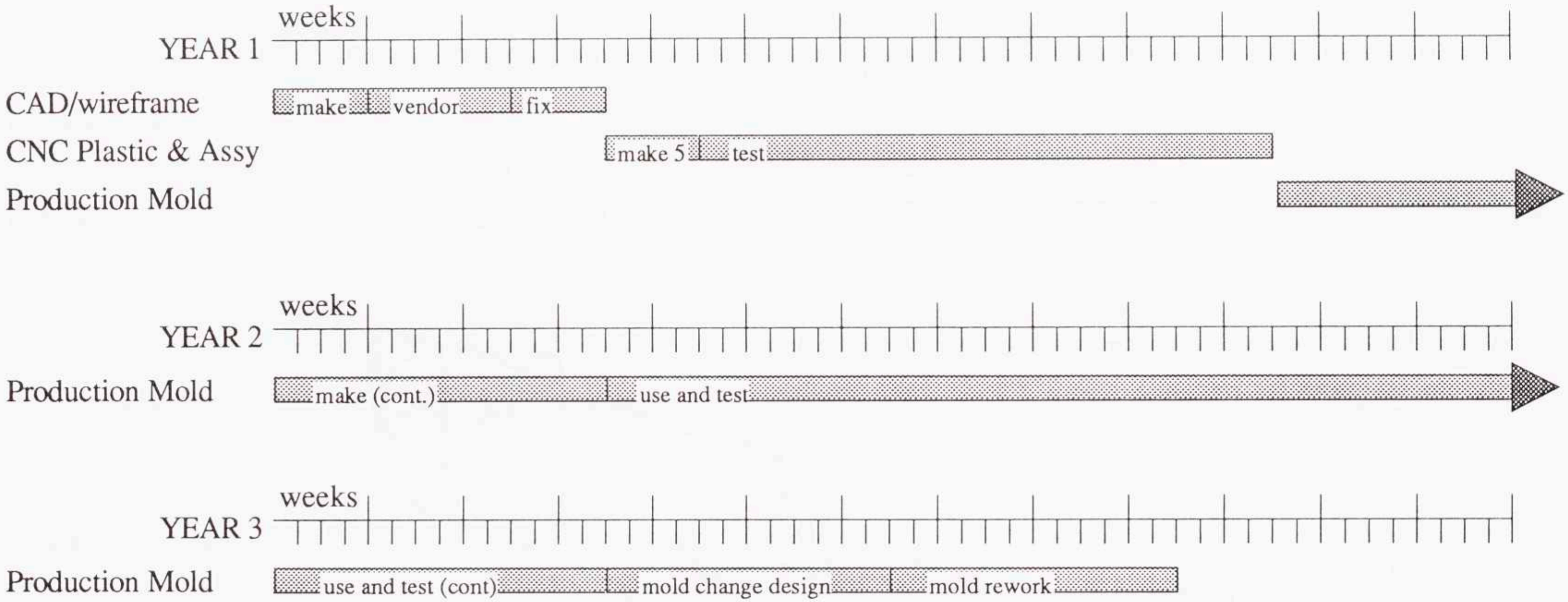
CAD/Wireframe -	Because the subsystem design was less defined than these parts, it was hard to define the subassembly interface.
CNC Plastic -	Stress testing showed problems, more interface problems. These were so serious that at one point the mold build was stopped.
Production Mold -	Snap features needed to be fixed. Part stiffness needed to be increased by adding ribs and increasing the part width. Pads added to ID of bearing holes to decrease bearing mounting hole tolerance.

113



An End Plate

PRE-FLIP STRUCTURE



PART HISTORY

Copier Document Feeder paper-handling part.

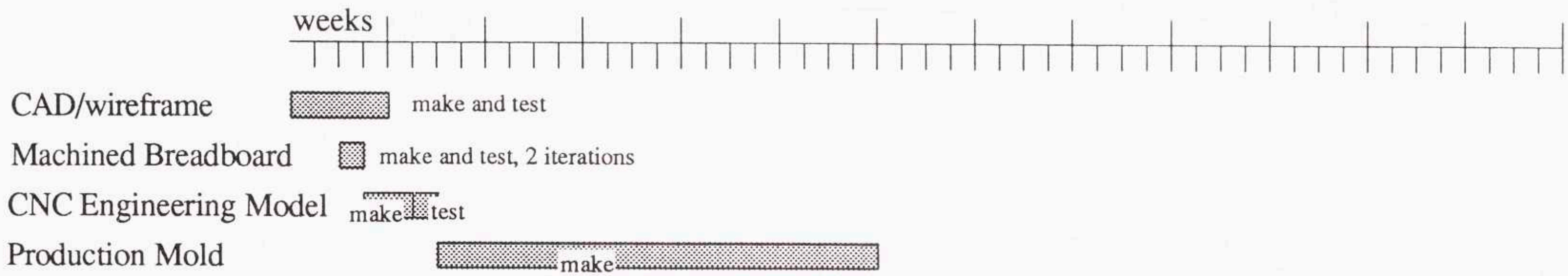
Prototyping Strategy:

CAD/Wireframe -	Designed without mold draft to show to mold maker, who then added the draft.
CNC Plastic and Assy -	5 functional parts for system tryout.

Problems Encountered:

CAD modified -	Draft interfered with a mating pin. Part of a wall had to be removed.
CNC Assy -	Field Service Group added view holes
Production Mold -	Too much paper flutter - added sheet metal ramps temporarily then designed ramps into the mold. There is still concern about the ejector pin marks on the paper handling surface.

COUNTER GEAR ASSY



PART HISTORY
Camera subassembly. Linkage that toggles to engage counter gear during automatic winding and rewinding. Part of complex internal camera frame.

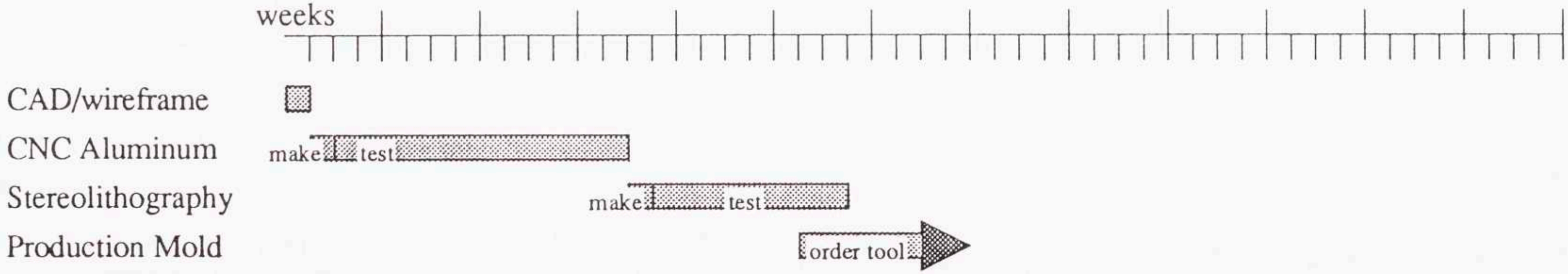
Prototyping Strategy:

CAD/wireframe -	Design and record.
Machine Breadboard -	Prove linkage concept. This prototype was added to the existing Gearplate Breadboard.
CNC Engineering Model -	Design was incorporated into the complete Camera Engineering Model
Production Mold -	

Problems Encountered:

Machine Breadboard -	Plastic cantilever spring did not work with linkage. Changed to 2-part lever and metal spring.
Production Mold -	Parts were not made at the time of this study.

PAPER DIVERTER



PART HISTORY

Copier Paper Feeder part which directs original sheets either into a tray or out of the feeder.

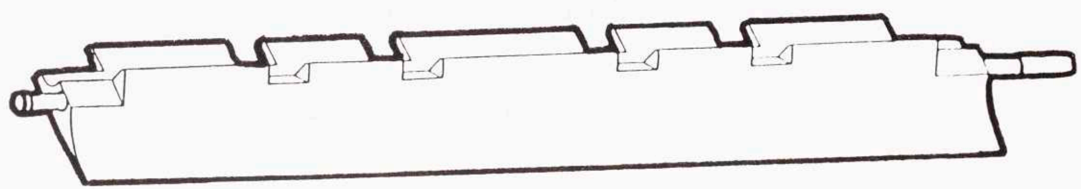
Prototyping Strategy:

CNC Aluminum -	Check function
Stereolithography -	Check geometry and fit

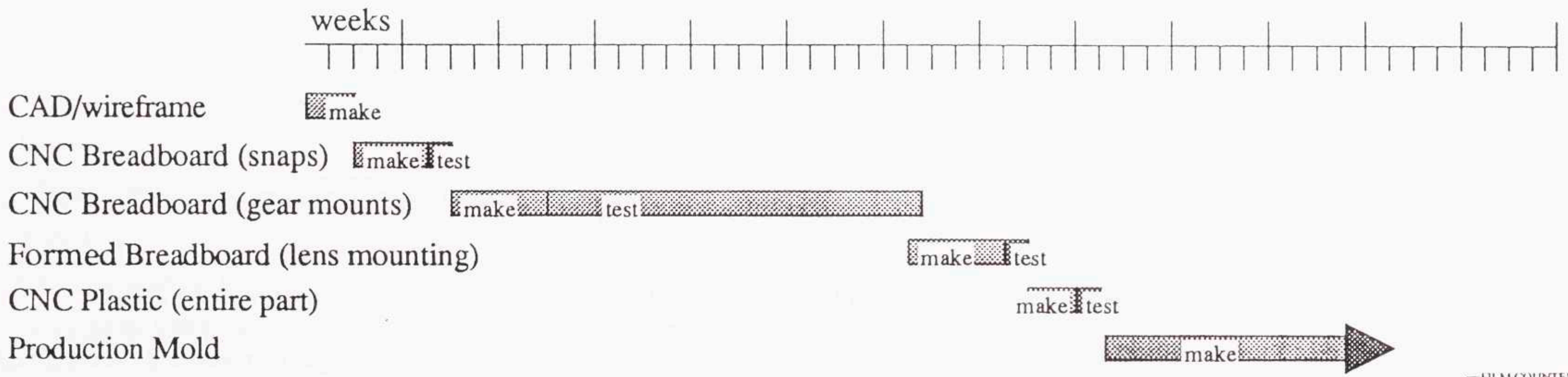
Problems Encountered:

CNC Aluminum -	Interference in several areas.
Stereolithography -	Found other interferences, unclear why these were not discovered with CNC Aluminum prototype. Also the Stereolithography part was difficult to test with because its surface was not as smooth and the radii had "jaggy" edges. The paper catches on these imperfections.

A Paper Diverter

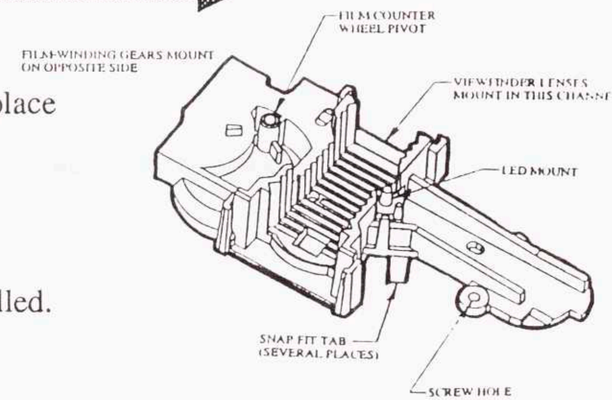


UPPER GEAR PLATE



PART HISTORY

Camera part . Multi-functional -- viewfinder, cover for film transport gear train. Snaps in place



An Upper Gear Plate

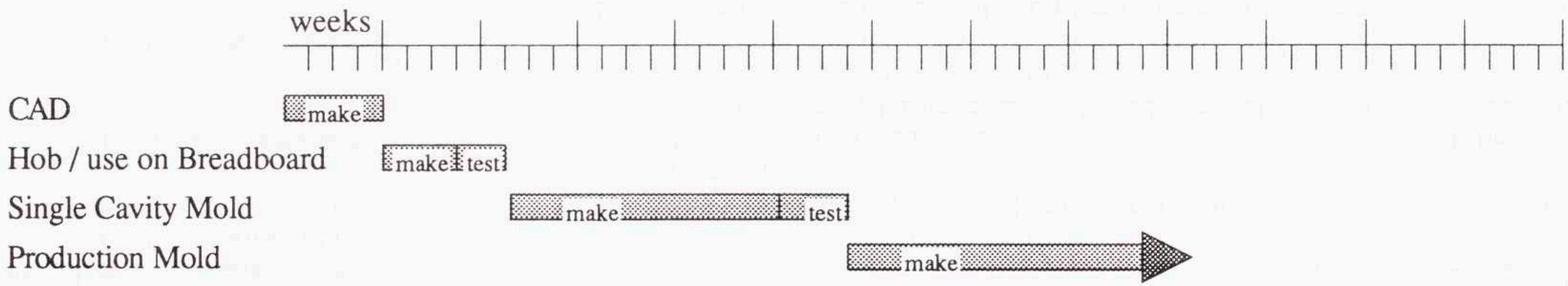
Prototyping Strategy:

CAD/Wireframe -	Design and Record.
CNC Breadboard (snaps)	Prove snaps worked.
CNC Breadboard (gear mounts)	Test gear transport with other components.
Formed Breadboard (lens mounting)	Insure that viewfinder lens stay securely installed.
CNC Plastic (entire part)	Test with entire assembly.
Production Mold	

Problems Encountered:

CNC Breadboard (snaps)	Snap beam was made thicker for more force.
CNC Breadboard (gear mounts)	Gear Post (a mating part) needed to be supported on the far side because it was deflecting; the deflection unmeshed the gears and caused them to slip.
Formed Breadboard (lens mounting)	Complex features could not be machined. A tool was made to melt the shape in the plastic. There was trouble with this process. It took several iterations to get the part right. Would use a single-cavity mold the next time.
CNC Plastic (entire part)	Assy interference when installing. Simple Breadboards did not model the installation. During gear stall, snaps pop off - replaced them with a screw - sometimes it's better not to have snaps.

CAMERA GEARS



PART HISTORY

Camera part . Drives film winder and picture counter display.
Cluster gears, pinion and large gear on same part with hole in middle to go on post.

Prototyping Strategy:

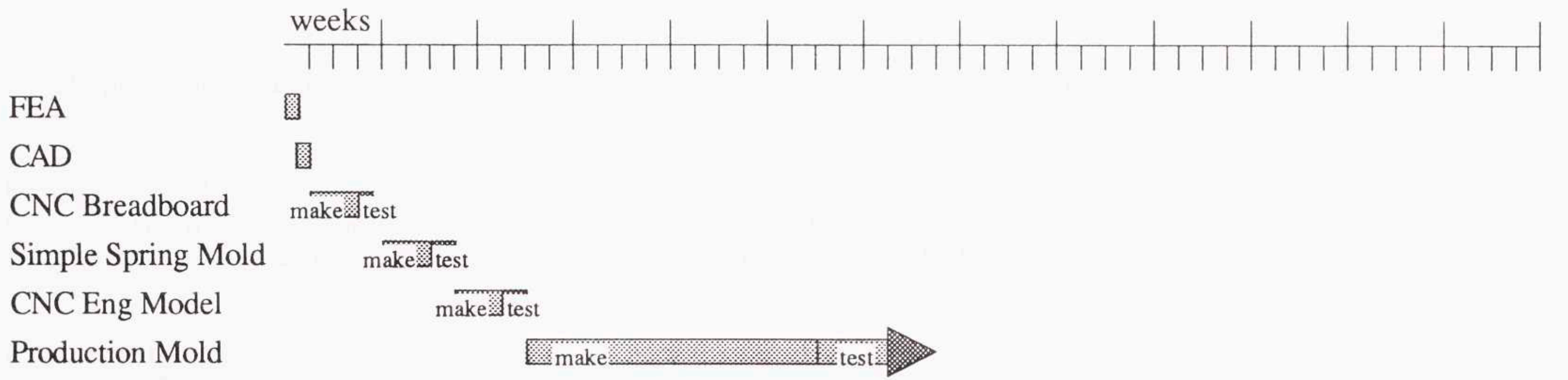
CAD	Iterations to check gear ratios and if they fit within housing constraints.
Hob from delrin/ use on BB*	Test film transport function.
Single Cavity Mold	Test gear sound.
Production Mold	No hurry, getting parts from single cavity.

Problems Encountered:

Hob from delrin/ use on BB	Concern about gear sound.
Single Cavity Mold	Gears warped - had to move ejector pin location.

* Same breadboard as Upper Gear Plate

DOOR CATCH AND SPRING



PART HISTORY

Camera part . Film door latch. Spring makes it pop open a little when latch is released.

Spring is a wavy, rod that compresses: 

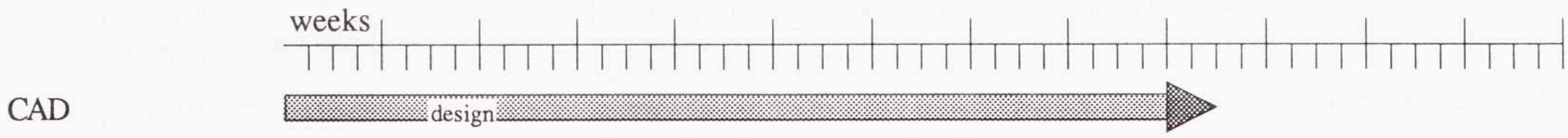
Prototyping Strategy:

FEA	Analysis done on simplified version of spring.
CAD	Design and Record.
CNC Breadboard	To test latch geometry.
Simple Spring Mold	Machined line in steel. First iteration designed undersize so that width of part (depth in mold) can be scaled up for proper spring force.
CNC Engineering Model	Test with entire assembly.
Production Mold	

Problems Encountered:

FEA	Need to build up material at bends.
CNC Breadboard	Machined springs broke, found reversing latch geometry made door easier to close and lock more secure.
Simple Spring Mold	Increased spring width.
CNC Engineering Model	Changed shape of release button to prevent accidental door opening.
Production Mold	Spring sticks in mold and the mold does not fill - added more draft and ejectors.

ELECTRONIC ENCLOSURE



PART HISTORY

Health Science Electronic Product Enclosure made of sheet metal and plastic (front and bezel)
Industrial Designers involved.

Problems Encountered:

The part has been stuck in redesign because there was poor communication between the Electrical Engineers, the Industrial designers, and the manufacturing group. The Electrical Engineers had some misunderstandings about injection molding - cost, time, and need for final design.

Note: Obviously this is one person's (the vendor's) opinion of the problems. I did not talk to the design group.

SPRING DRIVE

---- No time/effort info on this part because it was used to test Spray Metal, a new prototype process. ----

PART HISTORY

Copier Part. Mounts on shaft and rotates. A spring arm and nub engage and rotate a wiper shaft. Originally used a metal insert for the spring, but this was redesigned to be metal.

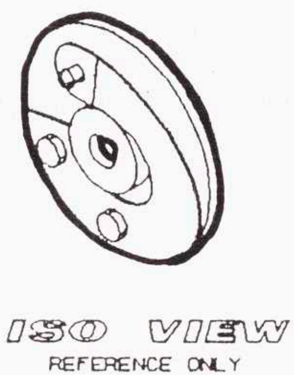
Prototyping Strategy:

CAD	Record, experiment with form.
CNC metal w/ riveted spring arm	Stress testing and assembly fit check.
Stereolithography	Visualize new plastic spring, master part for spray metal mold.
Spray Metal Mold	Parts for systems test, test plastic spring.
Production Mold	

Problems Encountered:

CNC metal	Shaft improperly aligned with nub.
Stereolithography	Could tell visually that plastic spring arm was too thin.

A Spring Drive



AIR TUBES

----- No effort/time info on this part ----

PART HISTORY

Health Science part for x-ray film processor. Tubes blow air through long slot in side. Desired more air flow than existing design provided.

Prototyping Strategy:

Sheet Metal	Experiment with slot and nozzle length.
Stereolithography	Show to molder.
Blow Mold	Production (similar to plastic bottle making; blow a bubble of plastic up in a mold). Slot cut in side.
Injection Mold	Make with two pieces and solvent bond.

Problems Encountered:

Sheet Metal	Adjust nozzle length
Blow Mold	Slot closed because of residual stress in part, temporarily changed process to heating the part before cutting

PLASTIC PART PROTOTYPING PROCESS DESCRIPTIONS

This appendix includes short descriptions of some of the processes used to prototype plastic parts.

Foam Core Assemblies

Foam core consists of a sheet of plastic foam sandwiched between two layers of cardboard. This material, a favorite with industrial designers, is available at most art and drawing supply stores. Foam core is easily cut with a razor blade. Assemblies can be made by hot gun gluing pieces together. To increase the realism of the prototype, renderings of product features can be pasted to the foam core with spray adhesive.

Foam core is good for appearance models, but only limited functioning.

Computer Aided Drawing (CAD)

CAD comes in three general flavors, 2D, 3D wireframe, and 3D solids.

2-Dimensional (2D) CAD is basically an electronic drafting board. Because the drawing storage is electronic, changes are more easily made than with paper and pencil. Paper output comes from pen, inkjet, or electrostatic plotters.

3-Dimensional (3D) Wireframe CAD electronically represents the surfaces of the part. The screen display shows the part as a network of lines describing the surface.

3-D Solids CAD electronically represents the entire volume of the part. The more powerful software packages show a fairly realistic looking part on the screen. 3-D Solids are better for testing the part fit in its assembly than wireframe. In addition, because the drawing commands are usually similar to machining operations (drilling, extruding, etc.), the designer has a better idea of how the part will be made. However, this may be problematic for molded parts with very complex geometries.

Both types of 3D CAD allow manipulation of the part and usually, with some translation, the files can be used for the computer analyses described below. They also require much more powerful computers that 2D packages.

Finite Element Analysis (FEA)

Finite Element Analysis (FEA) is a mathematical technique to determine the stresses within a part. The part is divided into small sections, sometimes called a mesh, and a force analysis is done on each section. The process is very calculation intensive and best done on a computer. 3D CAD models can be used as starting files for FEA.

Mold Flow Analysis

Mold Flow Analysis (MFA) is a mathematical technique to determine the molding properties of a plastic part. The properties include sink marks from uneven shrinking, warpage, air entrapment, overheating, the hot plastic flow path and ability of the plastic to fill the mold. The calculations are intensive and similar to FEA in that the part is divided into small sections. 3D CAD models can be used as starting files for MFA.

CNC Machined Material

Computer Numerical Controlled (CNC) machines are simply machine tools that use motors and feedback loops to control the tool movement. The CNC tool is programmed by the machinist or from a 3D CAD file. Multiple parts are made by repeating the program. Because of the power of the machine tools, a wide variety of materials (eg. metal, plastic, wood) can be used for the prototype.

A master machinist can make a CNC machined part look like it came out of a mold. But this part may not function exactly as the molded part would because it will not have the smooth skin from the mold surface. This can be problematic for parts that must act as springs; the tool marks can create stress concentrations.

Stereolithography (SLA)

Stereolithography (SLA) builds the part up in layers, using a 3D CAD file to generate the cross-sectional geometry. The part layers are created on top of each other by curing the part's cross-section in a thin surface layer of a photopolymer liquid. A laser is used for the curing energy. Mirrors point the laser beam to draw the cross-section. Because the laser is a point energy source, the entire surface can not be hardened in a reasonable time. Instead liquid is trapped in small areas, and the entire part is placed in an oven for final curing. 3D Systems (Valencia, CA) make the stereolithography system [Brown91, Deitz90, Leornard91, Miller91].

SLA parts can be made in a few hours. They are very good for examining the part's form.

The material used for SLA parts is brittle. This makes it difficult to use for functional parts. Because the layers have a finite width, curves and sloped areas of the part will have steps ("jaggies") from the corners of each layer. Post-processing can smooth the part's surface. The part can warp from the oven cure. There are efforts to improve the material properties and geometry tolerances of SLA parts.

Other Rapid Prototyping Layering Processes

Stereolithography was the first commercially available rapid prototyping layering process available. There are several others that use the same layering technique with some differences. Systems are available that make the entire cross-section in one flash similar to xerography and ones that use powders. The powder layers are solidified by laser sintering or an adhesive [Brown91, Deitz90, Leornard91, Miller91].

Although the performance of most of these processes is still unknown, expect some of them to challenge SLA's lead.

Rubber Molds

Part casting can be done in silicone rubber molds. A master part is made by machining, stereolithography, or other process. This master part is sprayed with a release agent and submersed in liquid silicone. After the silicone hardens (usually overnight), it is cut away from the master. The shell in the silicone is a mold that is filled with urethane or epoxy to make another part. The casting material also requires some cure time, usually overnight. Silicon can duplicate details as fine as a fingerprint or the ink on a dollar bill. Because the silicone is flexible, it will bend around undercuts when the mold is pulled away.

Parts from rubber molds can closely match production parts. There are some casting polymers that closely match injection molded plastic functionality and UL ratings. The mold is good for 10 to 200 parts depending on the part complexity and the care of the caster. The polymer cure time limits the speed of the process to a few parts per day.

Spray Metal Molds

Spray metal is like spray paint only louder, and it results in a harder and thicker surface. Two metal wires are melted and atomized by an electric arc between them. Pressurized air across the arc blows the metal onto a master part. The master can be a machined or a stereolithography part. The metal layer hardens and forms a shell over the master. The shell is split and, with each half backed by epoxy or a low-melting point metal, is used as an injection mold. Spray metal is capable of picking up very fine detail [Weiss90].

Spray metal molds are good for making simple geometry plastic parts. Unfortunately, the adhesion between the metal drops is not strong enough to stand up to the erosive forces of high pressure injection molding around sharp corners.

Other Non-traditional Molds

There are several types of processes to make molds besides the typical production process described below.

One technique involves casting the tool around a master. The casting material can be epoxy or a low-melting point metal. The master must be sprayed with a release agent and tough enough to withstand the casting process.

Another technique called Nickel Vapour Deposition makes a mold by depositing nickel on a master. The master must be made of a thermally conductive material because the nickel is condensed from vapors passed over it. The process can make molds of optical quality [Mirotech91].

There is also a technique that involves machining the mold cavity in aluminum or pre-hardened steel. It would seem that machining a cavity with a CNC tool should take the same amount of time as machining a prototype part. Yet, I saw many more machined prototypes than machined aluminum molds. Machining pre-hardened steel can be difficult because the hardness is often not uniform. An extra-hard spot can break the machining tool.

All of these tools are good for production type, injection molded parts. However the proper infrastructure is required to make the tools. There are some companies using proprietary processes to sell quick-turnaround, functional plastic prototypes for approximately 100 pieces [QMS91, ProDesign91].

Traditional Production Mold

A traditional production mold is machined into tool steel. The reverse of the part geometry is rough cut into annealed steel. EDM (see below) is used for complex features. The mold is heat treated. And, the final dimensions are achieved by grinding or EDM.

The mold releases the part by opening into two halves and pushing the part out with ejector pins. If there are any features (undercuts) in the side that would prevent the mold halves from opening, the mold must be designed so that side wall pulls away. This is called a side-draw or side-action.

There are two types of problems with molds. "Easy" ones require removing material from the mold (adding material to the part), "hard" ones require adding material to the mold (removing material from the part). Material is added to the mold by welding, which requires grinding to clean up the weld, or by cutting out part of the mold and replacing it with an insert. Both alternatives are time consuming

Frequently injection molds have more than one part cavity, i.e., the mold makes several parts at once. One prototyping technique is to make a single cavity in the mold blank and solve the problems there before making the other cavities.

Electric-Discharge Machining (EDM)

Electric-Discharge Machining (EDM) erodes metal with spark discharges between the tool and the workpiece. It is an important technology for making complex molds. The tool is machined from carbon in the shape of the part. These tools can be used to prototype the shape of the part.

DETAILS OF KJ DIAGRAM OF PROTOTYPING OBSERVATIONS

Because it is considered poor form to use indented format and square boxes for a KJ diagram, I have put the details of the KJ diagram shown in Figure 6 in this appendix. In this way, I can show the classical KJ groupings and major headings in the text and, at the same time, show the details here for those interested in the starting points.

Each group is separated from the other groups by the lines and boxes. The letters before the lower level points correspond to the letters in the groups in Figure 6.

WHAT CHARACTERIZES THE PROTOTYPING PROCESS?

EFFECTIVE PRODUCT DEVELOPMENT REQUIRES SOUND STRATEGY AND KNOWLEDGEABLE ENGINEERS

- ENGINEER'S KNOWLEDGE OF PART AND PROCESS CAPABILITIES ARE IMPORTANT FOR EFFECTIVE PRODUCT DEVELOPMENT.

Misunderstanding of Tool and Process Capabilities Put Part Development at Risk.

A. "D----" Project was delayed because engineers did not know limitations of plastic molding.

B. Designers use FEM few (<10) times a year.

C. Mistakes on prototypes can misguide you (eg. forget to include draft taper and you may not find an interference).

Experience with Prototyping Processes and with the Part's Prototyping History Makes Further Development More Efficient.

D. The problems that can be found at any given prototype stage depend on what was done earlier.

E. Past learning makes present prototyping more efficient.

F. Computer analysis is done by experts.

The Part's Performance is Affected by Itself and Its Environment.

G. Internal influences affect part performance (eg. material, process capability, number of critical features).

H. External influences affect part performance (eg. mating parts).

- PROTOTYPING STRATEGY DEPENDS ON PROCESSES AVAILABLE AND TIME TO BE SPENT ON EACH PROCESS.

There Is an Optimal Amount of Time, Not too Much or too Little to Most Effectively Use a Prototyping Process.

J. There are diminishing returns in finding problems over time when testing a prototype.

K. Have to do a certain amount of prototyping effort otherwise don't bother (it's worthless).

L. Engineer says: "Prototype of one is a prototype of none.", i.e. you need >1 prototypes.

Partial Prototype of the Easily Checked Parts Is Acceptable for Product Development if a Fast/Cheap Production Process Is Not Available.

M. If production tooling process is fast/cheap enough, it can be used to make prototypes (eg. laser printer).

N. During prototyping, like repair, one can check the problems that are easy to fix first.

P. Breadboards only do some of the functions of a part.

PROTOTYPE SEQUENCE IS CONSTRAINED BY DEPARTMENT PRACTICES AND PROCESS PREREQUISITES.

In Some Areas, the Prototyping Sequence Is Constrained by Department Policy, Rules, or Resources.

R. Camera Group has their own CNC machine shop.

S. In some products, parts are needed for regulatory test requirements.

T. LoB's use "Phases and Gates" for product development

.

CAD model is required to make hard tooling (and some prototypes).

REDUCING LEAD TIME IS IMPORTANT, BUT MAY LEAD TO INCREASED RISK.

Reducing Lead Time Is Important to Projects; Concurrent Engineering and Process Oriented Design Are Examples.

U. Surveys show that reducing lead time of prototypes is more important to engineers than reducing costs.

V. There are examples where parts are made more efficiently because they are designed with process in mind (eg. Front/Rear Frames).

W. There are examples of development time reduction with concurrent efforts.

Camera Group orders tools after CNC prototypes are built, but before fully testing.

EXPERIENCE WITH MOLD CHANGES AND PROTOTYPING PROCESSES INCREASE ENGINEERS' DESIRE TO PROTOTYPE.

There Are Some Limitations to CAD or Its Use That Allows the Designer to Miss Finding Some Problems.

X. Since the introduction of CAD, there are more engineering changes.

Y. SLA is used as a check of geometry in CAD 3D data base.

Z. Vacuum Hose Adjuster slot length problem was discovered with stereolithography, not CAD.

Changes Are Very Often Made to Mold, which Costs Money and Reliability.

AA. Reworking production tools affect their durability and reliability.

BB. Vendors make money on your mistakes by doing your rework.

CC. Changes are made in the mold in the majority of cases.

Engineers Want to Prototype to Be Sure They Did Not Make a Mistake.

DD. Engineer says: Object of prototyping is to reduce risk before building production mold.

EE. Kodak Engineers won't go directly to tool design.

FF. Designers express uncertainty about their work and a desire to check it with a prototype before committing to hard tooling.

Cost of field repair is highest of all categories of repair.

COMPUTER MODEL SIMULATIONS

This appendix is the code listing and results of one of the computer model simulations I ran. A portion of the output is also included.

This Appendix is divided into these parts:
- Data Structures
- Main Program (Shell and Engine)
- Make File
- Sample Output

Data Structures
(data_structures.h)

```
/*
   Data and data structures to be used in the Thesis program.

*/


#define NUM_OF_CONCERNS      4
#define NUM_OF_LEVELS        5
#define NUM_OF_PROCESSES     8
#define SEARCH_DEPTH         24
#define NUM_OF_BEST          5
#define MAX_LOG_LENGTH      50
#define THRESHOLD            0.1
#define STUDY_PERCENTAGE     0.5

static char  *concern_names[] = {
  "Function/Geometry",
  "Function/Material",
  "Producibility",
  "Marketability"
  };

enum {
  CONCERN_FUNCTION_GEOMETRY = 0,
  CONCERN_FUNCTION_MATERIAL,
  CONCERN_PRODUCIBILITY,
  CONCERN_MARKETABILITY
  }; /* CONCERN_FUNCTION_MATERIAL=1, CONCERN_PRODUCIBILITY=2,
etc */
```

```
static char  *process_names[] = {
  "CAD",
  "Mold Flow",
  "FEA",
  "SLA",
  "CNC/Plastic",
  "Rubber Mold",
  "Engineering Mold",
  "Production Mold"
  };

static char  *process_names_abv[] = {
  "CAD",
  "MFl",
  "FEA",
  "SLA",
  "CNC",
  "RbM",
  "EMd",
  "PMd"
  };

enum {
  PROCESS_CAD = 0,
  PROCESS_MOLD_FLOW,
  PROCESS_FEA,
  PROCESS_SLA,
  PROCESS_CNC_PLASTIC,
  PROCESS_RUBBER_MOLD,
  PROCESS_ENGINEERING_MOLD,
  PROCESS_PRODUCTION_MOLD
  };

/*   ABILITY MATRIX    */
/*   This one converts from process # to how each concern level changes
      (1 = perfect) */

static double  ability[ NUM_OF_CONCERNS ][ NUM_OF_PROCESSES ]
= {
  { 0.7, 0.0, 0.6, 0.7, 0.8, 0.8, 0.9, 1.0  }, /*
Function/Geometry */
  { 0.0, 0.8, 0.6, 0.0, 0.7, 0.8, 0.9, 1.0  }, /*
Function/Material */
  { 0.7, 0.8, 0.0, 0.8, 0.8, 0.9, 1.0, 1.0  }, /*
Producibility */
  { 0.4, 0.0, 0.0, 0.9, 0.9, 0.9, 1.0, 1.0  }  /*
Marketability */
};


/*    EFFECTIVE EFFORT VECTOR   */
/*   Maximum number of days that can be spent usefully on a
     given process including probable "quick fix" time
*/

static int   effective_time[ NUM_OF_PROCESSES ] =
    {  10, 2, 2, 2, 5, 8, 15, 40 };
```

```
/*    BUILD TIME VECTOR          */
/*   Delay time before the results of the process can be used. */

static int   build_time[ NUM_OF_PROCESSES ] =
    {  10, 2, 2, 2, 5, 8, 60, 100 };
/*
   Coupling factor between concern levels...

*/

static double  coupling[ NUM_OF_CONCERNS ][ NUM_OF_CONCERNS ]
= {
  { 1.0, 0.0, 0.0, 0.0  }, /* Function/Geometry */
  { 1.0, 0.0, 0.0, 0.0  }, /* Function/Material */
  { 1.0, 0.0, 0.0, 0.0  }, /* Producibility */
  { 1.0, 0.0, 0.0, 0.0  }  /* Marketability */
};



/*

  Data structures to be used in the program

*/


typedef struct {
  int     process;       /* Prototyping process used */
  int     study_time;    /* Study time used in days  */
  int     build_time;    /* Build time in days       */
} LOG_DATA, *LOG_PTR;


typedef struct {
  int      cumulative_effort;                 /* How many days spent    */
  double   concern_levels[ NUM_OF_CONCERNS ];
                                              /* Level of each concern */
  LOG_DATA log[ MAX_LOG_LENGTH ];             /* What we've done        */
  int      total_processes_used;              /* How many we've done    */
} STATE, *STATE_PTR;

static
STATE best_sequences[ NUM_OF_BEST ];
                    /* best finished sequences found sorted by time  */

static
STATE best_unfinished[ NUM_OF_BEST ];
                 /* best unfinished sequences sorted by concern sum */

static
found_all_flag;                     /* indicates best_sequence is full */
```

Shell and Engine Code
(main.c)

```
/*

  This is an exhaustive search strategy to find the best sequence
  of prototyping processes to use for the possible starting conditions.


*/

#include <stdio.h>
#include "data_structures.h"


/*  Function declarations   */

void engine_calc();
void find_sequences();
void time_sort();
int done_check();
void print_out();
void print_check();
void  print_concern ();
void concern_sort();
double composite_concern();


main()
{
  STATE   myState;
  int     i,j;
  int     c0,c1,c2,c3;         /* dummy concern categories  */

/* Print out Concern Category headers  */

  printf( "Depth of search = %d \n" , SEARCH_DEPTH );
  printf( "Study percentage = %6.2lf\n\n", STUDY_PERCENTAGE );
  printf( "Concern levels = \n" );
  for ( i = 0 ; i < NUM_OF_CONCERNS ; i++ )
      printf( "%s   ", concern_names[ i ] );
  printf("\n\n");

/* Nested loops to generate all possible concern levels  */

  for ( c0 = 0 ; c0 < NUM_OF_LEVELS ; c0++ ) {
    myState.concern_levels[ 0 ] = c0;

    for ( c1 = 0 ; c1 < NUM_OF_LEVELS ; c1++ ) {
      myState.concern_levels[ 1 ] = c1;

      for ( c2 = 0 ; c2 < NUM_OF_LEVELS ; c2++ ) {
        myState.concern_levels[ 2 ] = c2;

        for ( c3 = 0 ; c3 < NUM_OF_LEVELS ; c3++ ) {
          myState.concern_levels[ 3 ] = c3;
```

```
/* Initialize other state elements */

            myState.cumulative_effort = 0;
            myState.total_processes_used = 0;
            found_all_flag = 0;
            for (i = 0 ; i < NUM_OF_BEST ; i++) {
                  best_sequences[ i ].cumulative_effort = 9999;
                  best_sequences[ i ].total_processes_used = 0;
                  best_unfinished[ i ].total_processes_used = 0;
                  for ( j = 0; j < NUM_OF_CONCERNS; j++) {
                        best_unfinished[ i ].concern_levels[ j ] = 9999;
                        best_sequences[ i ].concern_levels[ j ] = 9999;
                  }
            }

/* run through all the possible prototype sequences and save the best */

        find_sequences( myState, SEARCH_DEPTH );

/* print starting values, and best processes */

      printf("Concern Levels = %d %d %d %d \n", c0, c1, c2, c3 );

      for  ( i = 0; i < NUM_OF_BEST ; i++)
            print_out( best_sequences[ i ] );

/* if not all were not good and finished, print out the three best
concern reducers  */

      if ( found_all_flag == 0) {
            printf("These are the unfinished sequences with lowest
concern levels:\n");
            for  ( i = 0; i < NUM_OF_BEST ; i++)
                  print_out( best_unfinished[ i ] );
      }

      printf( "\n" );

/* close 4 for loops and main program  */
        }
      }
    }
  }
}
```

```
/* this routine does a recursive search through all the possible  */
/* combinations of processes                                       */

void
find_sequences(state,depth)
STATE state;
int depth;
{
  STATE temp;
  int i;

  if ( found_all_flag == 1) {
      if ( state.cumulative_effort >
                                      best_sequences[ NUM_OF_BEST-1
].cumulative_effort )
      return;                           /* this sequence is already longer
*/
  }                               /* than the worst of the best list */

/* check if we have made it to the end */

  if ( done_check(state) == 1) {
      time_sort( state );
      return;
  }

  for ( i=0 ; i < NUM_OF_PROCESSES; i++) {
      temp = state;
      engine_calc( &temp, i );

      if ( depth > 0 )
             find_sequences(temp, depth-1);
      else {
             if ( done_check ( temp ) == 1 )
                    time_sort( temp );        /* put in good list  */
             else {
                    if (found_all_flag == 0 )
                           concern_sort( temp );
                                        /* put in unfinished list  */
             }
      }
  }
}



/* checks if sequence has made it below the threshold level  */
int
done_check( s )
STATE s;
{
       int i;
       for( i = 0; i < NUM_OF_CONCERNS; i++ ) {
              if ( s.concern_levels[ i ] > THRESHOLD )
                     return( 0 );
       }
       return( 1 );
}
```

```
/* checks a sequence against the three best so far by time and sets
found_all_flag if all full  */
void
time_sort( s )
STATE s;
{
  int i,j;

  for ( i = 0 ; i < NUM_OF_BEST ; i++) {

      if (s.cumulative_effort < best_sequences[ i ].cumulative_effort) {

            for ( j = NUM_OF_BEST-1 ; j > i ; j--)
                  best_sequences[ j ] = best_sequences[ j-1 ];

            best_sequences[ i ] = s;

            if ( found_all_flag == 0) {
                  if (done_check( best_sequences[ NUM_OF_BEST - 1 ] ) ==
1 )
                        found_all_flag = 1;
            }

            break;
      }
  }
}



/* checks a sequence against the three best unfinished and sorts by
composite concern (sum) */
void
concern_sort( s )
STATE s;
{
  double csum;
  int i,j;

  csum = composite_concern( s );

  for ( i = 0 ; i < NUM_OF_BEST ; i++) {

      if ( csum < composite_concern( best_unfinished[ i ]) ) {

            for ( j = NUM_OF_BEST-1 ; j > i ; j--)
                  best_unfinished[ j ] = best_unfinished[ j-1 ];

            best_unfinished[ i ] = s;
            break;
      }
  }
}
```

```
/* adds up all the concern levels  */
double
composite_concern( s )
STATE s;
{
  double csum;
  int i;

  csum = 0.0;

  for ( i = 0 ; i < NUM_OF_CONCERNS ; i++ )
      csum += s.concern_levels[ i ];

  return( csum);
}



/* this prints out the best sequences for the concern levels   */
void
print_out( s )
STATE s;

{
  int i,j;

      printf("%d ", s.cumulative_effort );

      for ( j =  0 ; j < s.total_processes_used ; j++ )
            printf("%s ", process_names_abv[ s.log[ j ].process ] );

      print_concern ( s );
      printf("\n");

  }




/* print out current  set of processes */
void
print_check( s )
STATE s;
{
      int i;
      printf(" %d",s.total_processes_used);
      for ( i = 0; i < s.total_processes_used; i++ )
            printf("%s ", process_names_abv[ s.log[ i ].process ] );
      printf("\n");
}
```

```
/* print concern levels  */
void
print_concern ( s )
STATE s;
{
      int i;
      for ( i = 0; i < NUM_OF_CONCERNS; i++ ) {
            printf(" %6.2lf",s.concern_levels[ i ] );
      }
}


/**********************************************************/
/*
/*     Basic engine.  Pass this a nice fresh STATE and
/*     a process to use (perhaps from a heuristic) and it
/*     will figure out which process to use and the effect.
/*
/**********************************************************/


void
engine_calc( state_p, process_number)
STATE_PTR state_p;
int    process_number;
{
  double actual_study_percentage;
  double max_delta_concern[ NUM_OF_CONCERNS ];
  double actual_delta_concern[ NUM_OF_CONCERNS ];
  int    this_build_time;
  int    this_study_time;
  int    i;

/* increment effort ,ie, time   */
  this_build_time = build_time[ process_number ];
  this_study_time = STUDY_PERCENTAGE * effective_time[ process_number ];

  state_p->cumulative_effort += this_build_time + this_study_time;

  /* Now calculate the change in concern levels */

  actual_study_percentage =
            this_study_time / effective_time[ process_number ];
  if ( actual_study_percentage > 1.0 )
    actual_study_percentage = 1.0;  /* After effective_effort,      */
                               /* diminishing returns keeps you  */
                               /* from learning more           */


  /* First calculate how far they COULD change */

  for ( i = 0 ; i < NUM_OF_CONCERNS ; i++ ) {
    max_delta_concern[ i ] = state_p->concern_levels[ i ]
                                  * ability[ i ][ process_number ];
    actual_delta_concern[ i ] = max_delta_concern[i] *
actual_study_percentage;
  }
```

```
  /* If coupling effects are desired, then add formulas in thesis text
here*/

  /* Now put this back into the old state, ie, add delta to original */

  for ( i = 0 ; i < NUM_OF_CONCERNS ; i++ ) {
    state_p->concern_levels[i] -= actual_delta_concern[i];

    if ( state_p->concern_levels[i] < 0.0 )
      state_p->concern_levels[i] = 0.0;         /* limit bottom to zero
*/
  }

  /*  Log the data away */

  state_p->log[ state_p->total_processes_used ].process =
process_number;
  state_p->log[ state_p->total_processes_used ].study_time =
this_study_time;
  state_p->log[ state_p->total_processes_used ].build_time =
this_build_time;

  state_p->total_processes_used++;
}
```

Make file
(Makefile)

```
CFLAGS        = -g
CC            =  cc

OBJS          = main.o
SRCS          = main.c

all:    thesis

thesis: ${OBJS}
        cc -o thesis ${OBJS} -lm

depend:
        makedepend -- $(CFLAGS) -- $(SRCS)

.c.o:
        cc -c ${CFLAGS} $*.c

# DO NOT DELETE THIS LINE -- make depend depends on it.

main.o: /usr/include/stdio.h data_structures.h
```

Sample Results
This is part of the output from the computer code above.

It begins by stating the Concern categories. Then there is a section for each starting set of Concern Levels. Each section contains the following:

- The starting concern levels
- The five best sequences found.
  - The time for each sequence.
  - The order of processes in each sequence.
  - The final set of concern values.

Notice that only three processes are in all the best sequences. These are SLA (Stereolithography), MFl (Mold Flow Analysis), and FEA (Finite Element Analysis). These were the ones with the shortest Build Time and Effective Study Time. See *Results of Computer Model Simulations* for discussion.

```
Concern levels =
Function/Geometry   Function/Material   producibility   Marketability

Concern Levels = 0 0 0 1
12 SLA SLA SLA SLA    0.00   0.00   0.00   0.09
15 MFl SLA SLA SLA SLA    0.00   0.00   0.00   0.09
15 FEA SLA SLA SLA SLA    0.00   0.00   0.00   0.09
15 SLA MFl SLA SLA SLA    0.00   0.00   0.00   0.09
15 SLA FEA SLA SLA SLA    0.00   0.00   0.00   0.09

Concern Levels = 0 0 0 2
18 SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
21 MFl SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
21 FEA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
21 SLA MFl SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
21 SLA FEA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06

Concern Levels = 0 0 0 3
18 SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.08
21 MFl SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.08
21 FEA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.08
21 SLA MFl SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.08
21 SLA FEA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.08

Concern Levels = 0 0 0 4
21 SLA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
24 MFl SLA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
24 FEA SLA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
24 SLA MFl SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06
24 SLA FEA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.00   0.06

Concern Levels = 0 0 1 0
15 MFl MFl MFl MFl MFl    0.00   0.00   0.08   0.00
15 MFl MFl MFl MFl SLA    0.00   0.00   0.08   0.00
15 MFl MFl MFl SLA MFl    0.00   0.00   0.08   0.00
15 MFl MFl MFl SLA SLA    0.00   0.00   0.08   0.00
15 MFl MFl SLA MFl MFl    0.00   0.00   0.08   0.00
```

```
Concern Levels = 0 0 1 1
15 MFl SLA SLA SLA SLA    0.00   0.00   0.08   0.09
15 SLA MFl SLA SLA SLA    0.00   0.00   0.08   0.09
15 SLA SLA MFl SLA SLA    0.00   0.00   0.08   0.09
15 SLA SLA SLA MFl SLA    0.00   0.00   0.08   0.09
15 SLA SLA SLA SLA MFl    0.00   0.00   0.08   0.09

Concern Levels = 0 0 1 2
18 SLA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.06
21 MFl SLA SLA SLA SLA SLA SLA    0.00   0.00   0.03   0.06
21 FEA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.06
21 SLA MFl SLA SLA SLA SLA SLA    0.00   0.00   0.03   0.06
21 SLA FEA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.06

Concern Levels = 0 0 1 3
18 SLA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.08
21 MFl SLA SLA SLA SLA SLA SLA    0.00   0.00   0.03   0.08
21 FEA SLA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.08
21 SLA MFl SLA SLA SLA SLA SLA    0.00   0.00   0.03   0.08
21 SLA FEA SLA SLA SLA SLA SLA    0.00   0.00   0.05   0.08



------------------------------------------------------------
------------------- part of output removed -----------------
------------------------------------------------------------



Concern Levels = 0 1 2 4
36 MFl MFl MFl MFl MFl SLA SLA SLA SLA SLA SLA SLA    0.00   0.08   0.00
0.06
36 MFl MFl MFl MFl FEA SLA SLA SLA SLA SLA SLA SLA    0.00   0.09   0.01
0.06
36 MFl MFl MFl MFl SLA MFl SLA SLA SLA SLA SLA SLA    0.00   0.08   0.00
0.06
36 MFl MFl MFl MFl SLA FEA SLA SLA SLA SLA SLA SLA    0.00   0.09   0.01
0.06
36 MFl MFl MFl MFl SLA SLA MFl SLA SLA SLA SLA SLA    0.00   0.08   0.00
0.06

Concern Levels = 0 1 3 0
21 MFl MFl MFl MFl MFl MFl MFl    0.00   0.03   0.08   0.00
21 MFl MFl MFl MFl MFl MFl SLA    0.00   0.05   0.08   0.00
21 MFl MFl MFl MFl MFl SLA MFl    0.00   0.05   0.08   0.00
21 MFl MFl MFl MFl MFl SLA SLA    0.00   0.08   0.08   0.00
21 MFl MFl MFl MFl SLA MFl MFl    0.00   0.05   0.08   0.00

Concern Levels = 0 1 3 1
27 MFl MFl MFl MFl MFl SLA SLA SLA SLA    0.00   0.08   0.03   0.09
27 MFl MFl MFl MFl FEA SLA SLA SLA SLA    0.00   0.09   0.05   0.09
27 MFl MFl MFl MFl SLA MFl SLA SLA SLA    0.00   0.08   0.03   0.09
27 MFl MFl MFl MFl SLA FEA SLA SLA SLA    0.00   0.09   0.05   0.09
27 MFl MFl MFl MFl SLA SLA MFl SLA SLA    0.00   0.08   0.03   0.09
```

```
Concern Levels = 0 1 3 2
33 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.06
33 MF1 MF1 MF1 MF1 FEA SLA SLA SLA SLA SLA SLA    0.00    0.09    0.02
0.06
33 MF1 MF1 MF1 MF1 SLA MF1 SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.06
33 MF1 MF1 MF1 MF1 SLA FEA SLA SLA SLA SLA SLA    0.00    0.09    0.02
0.06
33 MF1 MF1 MF1 MF1 SLA SLA MF1 SLA SLA SLA SLA    0.00    0.08    0.01
0.06

Concern Levels = 0 1 3 3
33 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.08
33 MF1 MF1 MF1 MF1 FEA SLA SLA SLA SLA SLA SLA    0.00    0.09    0.02
0.08
33 MF1 MF1 MF1 MF1 SLA MF1 SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.08
33 MF1 MF1 MF1 MF1 SLA FEA SLA SLA SLA SLA SLA    0.00    0.09    0.02
0.08
33 MF1 MF1 MF1 MF1 SLA SLA MF1 SLA SLA SLA SLA    0.00    0.08    0.01
0.08

Concern Levels = 0 1 3 4
36 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.06
36 MF1 MF1 MF1 MF1 FEA SLA SLA SLA SLA SLA SLA SLA    0.00    0.09    0.01
0.06
36 MF1 MF1 MF1 MF1 SLA MF1 SLA SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.06
36 MF1 MF1 MF1 MF1 SLA FEA SLA SLA SLA SLA SLA SLA    0.00    0.09    0.01
0.06
36 MF1 MF1 MF1 MF1 SLA SLA MF1 SLA SLA SLA SLA SLA    0.00    0.08    0.01
0.06

Concern Levels = 0 1 4 0
24 MF1 MF1 MF1 MF1 MF1 MF1 MF1 MF1    0.00    0.02    0.07    0.00
24 MF1 MF1 MF1 MF1 MF1 MF1 MF1 SLA    0.00    0.03    0.07    0.00
24 MF1 MF1 MF1 MF1 MF1 MF1 SLA MF1    0.00    0.03    0.07    0.00
24 MF1 MF1 MF1 MF1 MF1 MF1 SLA SLA    0.00    0.05    0.07    0.00
24 MF1 MF1 MF1 MF1 MF1 SLA MF1 MF1    0.00    0.03    0.07    0.00

-------------------------------------------------------------
------------------- part of output removed ------------------
-------------------------------------------------------------

Concern Levels = 0 2 2 2
36 MF1 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA SLA SLA SLA    0.00    0.09    0.00
0.06
36 MF1 MF1 MF1 MF1 MF1 SLA MF1 SLA SLA SLA SLA SLA    0.00    0.09    0.00
0.06
36 MF1 MF1 MF1 MF1 MF1 SLA SLA MF1 SLA SLA SLA SLA    0.00    0.09    0.00
0.06
36 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA MF1 SLA SLA SLA    0.00    0.09    0.00
0.06
36 MF1 MF1 MF1 MF1 MF1 SLA SLA SLA SLA MF1 SLA SLA    0.00    0.09    0.00
0.06
```

```
Concern Levels = 0 2 2 3
36 MFl MFl MFl MFl MFl MFl SLA SLA SLA SLA SLA SLA    0.00    0.09    0.00
0.08
36 MFl MFl MFl MFl MFl SLA MFl SLA SLA SLA SLA SLA    0.00    0.09    0.00
0.08
36 MFl MFl MFl MFl MFl SLA SLA MFl SLA SLA SLA SLA    0.00    0.09    0.00
0.08
36 MFl MFl MFl MFl MFl SLA SLA SLA MFl SLA SLA SLA    0.00    0.09    0.00
0.08
36 MFl MFl MFl MFl MFl SLA SLA SLA SLA MFl SLA SLA    0.00    0.09    0.00
0.08

Concern Levels = 0 2 2 4
39 MFl MFl MFl MFl MFl MFl SLA SLA SLA SLA SLA SLA SLA    0.00    0.09
0.00   0.06
39 MFl MFl MFl MFl MFl SLA MFl SLA SLA SLA SLA SLA SLA    0.00    0.09
0.00   0.06
39 MFl MFl MFl MFl MFl SLA SLA MFl SLA SLA SLA SLA SLA    0.00    0.09
0.00   0.06
39 MFl MFl MFl MFl MFl SLA SLA SLA MFl SLA SLA SLA SLA    0.00    0.09
0.00   0.06
39 MFl MFl MFl MFl MFl SLA SLA SLA SLA MFl SLA SLA SLA    0.00    0.09
0.00   0.06

Concern Levels = 0 2 3 0
21 MFl MFl MFl MFl MFl MFl MFl    0.00    0.06    0.08    0.00
21 MFl MFl MFl MFl MFl MFl SLA    0.00    0.09    0.08    0.00
21 MFl MFl MFl MFl MFl SLA MFl    0.00    0.09    0.08    0.00
21 MFl MFl MFl MFl SLA MFl MFl    0.00    0.09    0.08    0.00
21 MFl MFl MFl SLA MFl MFl MFl    0.00    0.09    0.08    0.00

Concern Levels = 0 2 3 1
30 MFl MFl MFl MFl MFl MFl SLA SLA SLA SLA    0.00    0.09    0.02    0.09
30 MFl MFl MFl MFl MFl SLA MFl SLA SLA SLA    0.00    0.09    0.02    0.09
30 MFl MFl MFl MFl MFl SLA SLA MFl SLA SLA    0.00    0.09    0.02    0.09
30 MFl MFl MFl MFl MFl SLA SLA SLA MFl SLA    0.00    0.09    0.02    0.09
30 MFl MFl MFl MFl MFl SLA SLA SLA SLA MFl    0.00    0.09    0.02    0.09


-------------------------------------------------------------
------------------- part of output removed ------------------
-------------------------------------------------------------


Concern Levels = 0 3 1 3
39 MFl MFl MFl MFl MFl MFl MFl SLA SLA SLA SLA SLA SLA    0.00    0.08
0.00   0.08
39 MFl MFl MFl MFl MFl MFl FEA SLA SLA SLA SLA SLA SLA    0.00    0.10
0.00   0.08
39 MFl MFl MFl MFl MFl MFl SLA MFl SLA SLA SLA SLA SLA    0.00    0.08
0.00   0.08
39 MFl MFl MFl MFl MFl MFl SLA FEA SLA SLA SLA SLA SLA    0.00    0.10
0.00   0.08
39 MFl MFl MFl MFl MFl MFl SLA SLA MFl SLA SLA SLA SLA    0.00    0.08
0.00   0.08
```

```
Concern Levels = 0 3 1 4
42 MFl MFl MFl MFl MFl MFl MFl SLA SLA SLA SLA SLA SLA SLA    0.00
0.08   0.00   0.06
42 MFl MFl MFl MFl MFl MFl FEA SLA SLA SLA SLA SLA SLA SLA    0.00
0.10   0.00   0.06
42 MFl MFl MFl MFl MFl MFl SLA MFl SLA SLA SLA SLA SLA SLA    0.00
0.08   0.00   0.06
42 MFl MFl MFl MFl MFl MFl SLA FEA SLA SLA SLA SLA SLA SLA    0.00
0.10   0.00   0.06
42 MFl MFl MFl MFl MFl MFl SLA SLA MFl SLA SLA SLA SLA SLA    0.00
0.08   0.00   0.06

Concern Levels = 0 3 2 0
21 MFl MFl MFl MFl MFl MFl MFl    0.00   0.08   0.06   0.00
21 MFl MFl MFl MFl MFl MFl FEA    0.00   0.10   0.09   0.00
21 MFl MFl MFl MFl MFl FEA MFl    0.00   0.10   0.09   0.00
21 MFl MFl MFl MFl FEA MFl MFl    0.00   0.10   0.09   0.00
21 MFl MFl MFl FEA MFl MFl MFl    0.00   0.10   0.09   0.00

-------------------------------------------------------------
----------------- computer output stopped -------------------
-------------------------------------------------------------
```

< THE END >


ANOTHER
MESSY DESK
PRODUCTION