ABSTRACT

The overall object of this project is the computerization of the purchasing department of a large marine fabrication company. The company fabricates large structures and systems for marine use in widely scattered locations around the world, but virtually all of its purchasing is done from a centralized office at the company headquarters.

Requisitions from a fabrication project are received by the purchasing department, which determines the best source for the materials and produces a purchase order to be sent to the vendor. Purchasing is also responsible for ensuring that the materials are shipped and received within a reasonable period of time: this function is performed by the "expediters".

It is common to order a large quantity of a given material and specify that it is to be broken down into a number of different items, each with its own shipping date and destination. It is necessary to track and note receipt of these shipments elsewhere.

Purchasing is also burdened with the production of a variety of reports concerning the cost of materials for the multitude of systems, subsystems and parts which make up a complete project.

This Master's Project will demonstrate the development and implementation of a software package which will allow these functions to be performed on a user-friendly, interactive computer system.
Purchase order information will be entered into the system, which will prepare and print the finished purchase orders. The system will maintain the relationships between the P.O.'s, the shipments and the fabrication projects within its data base. In the same manner, it will maintain up-to-date statistics on the costs associated with material requirements for the project and its systems, subsystems, and parts. The statistics, and the relationships among them will be used to create a variety of reports for corporate managers at all levels. At the same time, much of this information will be immediately available at a terminal for day-to-day use.
TABLE OF CONTENTS

INTRODUCTION

CHAPTER I. ANALYSIS AND DESIGN

CHAPTER II. DOCUMENTATION

CHAPTER III. IMPLEMENTATION

CHAPTER IV A MAJOR POST-IMPLEMENTATION CHANGE

CHAPTER V. RESULTS

APPENDIX A. RECORD LAYOUTS

APPENDIX B. STRUCTURE CHARTS

APPENDIX C. USER INSTRUCTIONS

APPENDIX D. SOURCE LISTING
INTRODUCTION

The overall objective of this project is the computerization of a large marine fabrication company. It represents an actual project undertaken for a particular company, and as such it shows the influence of that particular company's needs. The company fabricates large structures and systems in widely scattered locations around the world, but virtually all of its purchasing is done from a relatively small office at the company's headquarters. The computer system in question was being purchased in anticipation of the construction of a mammoth offshore drilling rig at a construction site halfway around the world.

A mobile drilling unit is a very large and complex structure which is assembled from a number of lesser systems, each of which is in turn from a varying number of subsystems. These subsystems are of course assembled from parts: hundreds of thousands of different parts go into each finished rig.

Since this was to be an international venture of considerable scope, the burden to be placed on the purchasing department promised to be a heavy one.

The flow of information in a typical material acquisition is as follows. First, a requisition is received for the purchase of something. This requisition will spell out what the specifications are, what is to be delivered, and by what date it must arrive. The requisition is given to the purchasing department, and finds its way into the hands of a buyer. Each buyer has his or her own area of expertise, and is well acquainted with the purveyors of whatever it is that is being requisitioned. Some-
times there are long-term contracts governing the purchase of certain materials, and sometimes there aren't. The buyer will determine which vendor the material will be purchased from, the terms of purchase, and the price. This information, along with the requisition, are then handed to the typist, who must then prepare and type the purchase order. The purchase order must then be reviewed, approved, and signed by the person with the appropriate authority. Once sent, the items ordered enter the realm of the expeditor. It is the expeditor's job to ensure that the material is shipped at the time and in the manner specified, that it doesn't get lost in transit, and that it is sent to the proper destination. The expeditor must keep track of shipments which have been delayed, the reasons for the delay, and so on. A large portion of the materials are shipped to a company facility, where they are received and examined before being sent to the final destination. Receipt of the goods is the final step in the process, and one which interfaces directly with the purchasing department. Receiving personnel receive the shipment, note how much was received and the date of receipt. Both overages and under shipments are common. Receiving must often repack the goods and ship them on to their final destination, so they need to know for whom it was ordered. Purchasing, of course, needs to know when the goods are received so that payment can be authorized. Documentation on the received goods is often lacking, because many of them are drop shipped. And, as is so often the case, the job isn't finished until the paperwork's done: paper must be sent back to purchasing to document receipt.
INTRODUCTION

Purchasing is also burdened with a flurry of requests from other departments for reports on such things as projected cash flow, historical costing information, etc.

From discussion with the purchasing department representatives (hereafter referred to as "the user"), certain specific requirements emerged. One requirement was that the current flow of information be disrupted as little as possible and that the computer duplicate existing documents and procedures to the greatest extent practicable. Another requirement was that a single line item on a purchase order be broken down into a variable number of shipments, and that partial shipments must be allowed after the order has been placed. In addition, the particular subsystem for which it was ordered must be known.

Another requirement was that costing data be maintained on the different systems and subsystems, and that some basic statistics also be kept on the amounts of individual parts both purchased or on order.

Since a complete system was being purchased (hardware and software at a single inclusive price) the cost of the total system was an important factor. A second, and equally important factor was that the system had to be implemented and installed within four months. A supplementary period of two months was allowed for post installation modifications and debugging.

These two factors dominated all others in the design and implementation of the system. The cost factor limited the system to an eight-user machine with a total of 52 megabytes of hard disk storage: the CADO 20/28 with Century Marksman drives. In light of
the programming resources available for the project, the time factor meant that the software would be very much of a one shot situation: any attempt at a re-design after the first two or three weeks might cause fatal delays.

It should be explained at this point that the CADO system uses a proprietary operating system and programming language. The operating system is a multi-user, multi-tasking system designed for high performance. It allows interfacing between data processing, word processing, and a data base manipulation language called Just Ask. The programming language is CADOOL, which is a form of compiler BASIC which has been extensively modified to achieve high performance in a small user partition. The use of CADOL dictates some of the particulars of record layouts and control structures. Both the operating system and the programming language are peculiar to CADO.

The first concept adopted in the design of the software was that of conceptual simplicity. Any data structures to be adopted or programming techniques to be used had to pass this test. One reason for this policy was to attempt to limit the possible number of variables encountered in testing and to make it easier to debug. Complex control structures in the code which might exhibit unexpected behavior when modified or enhanced were rejected. The use of "clever" manipulations to increase performance were avoided. The desired result was code that was easy to use and simple to debug.

Another design criteria was that expansion room be left in
INTRODUCTION

the records and in the structure so that future enhancements
would be easier to accomplish. The system couldn't be set up
around rigid tolerances which might necessitate major rewrites or
restructuring in the event of future changes.

Once the philosophical mood was set, the next step in the
design was to determine the functional data flow. This proved to
be a relatively easy task, since the computer system was to emu-
late a paper system that was well established and defined.

Next came the more difficult task of defining the structure
that the data was to take. The data associated with parts and
vendors was independent of other relationships, and so the parts
and vendors were simply assigned their own keys and individual
records. The purchase order number was made the foundation of the
purchase order-related data. The purchase order was made up of
three separate kinds of information: the top, which detailed ele-
ments common to all line items, the line item, which detailed
what was being purchased, and the shipments, which were dependent
upon a particular line item. The breakdown of the rig into system
and subsystem turned out to be easy: the information kept on
each was identical, so the record layout was the same. It was
merely a matter of showing the relationships among the records.
Once the essential data was laid out, the supporting data(messa-
ges, textual descriptions and the like) was detailed.

A combination of interactive and batch processing was used.
All information which was to be input by operators would, of
course, be entered by interactive user-friendly programs. The
INTRODUCTION

purchasing information, once entered and edited, would be
"posted" to the cost fields of the system, parts, and vendor re-
cords. Reports were to be generated at will.

The flow of the screen logic was then laid out. A menu-dri-
ven system was envisioned, and the essential screens were pre-
pared in rough form.

At this point, another philosophical rule was formulated. The
user had to approve the system at its various stages. The reason
for this rule was to completely eliminate any possible misunder-
standings or recriminations on delivery of the system. There had
to be some definition and limits to the system being written. The
specifications were typed up and submitted. Although necessary,
this rule had the effect of "locking in" a design, especially
when viewed in light of the time constraints. Once it had been
submitted, the temptation to "start over" was effectively re-
moved. This, in turn, made the use of simple concepts even more
important. The emphasis here being to minimize obstacles during
implementation.

The user requested that an implementation schedule be inclu-
ded. The effects of the implementation schedule were for the most
part beneficial: they added that sense of urgency so necessary
for maintaining motivation.

The most important criteria used in the implementation phase
was that all code produced be easy to maintain. Structure, modu-
ularity, and documentation were stressed. The second governing
criteria was that the resulting programs be easy to use. Every
attempt was made to keep the programs user-friendly and simple.
INTRODUCTION

The implementation schedule was set up so that each stage laid a foundation of functions which were complete at that point. The idea was to write the routines which governed entry and editing first, with other, more complex functions coming later. As each stage was completed, the associated functions would be completely tested. In this way, some parts of the system were usable long before the complete system was ready. One of the objectives of this approach was to give the user the opportunity to install his equipment and begin building a parts and vendors data base before the end of the four month deadline. This would greatly reduce the time needed to bring the system up after completion of all of the software. It also gave the user the chance to make some "mid-course" corrections should serious oversights in the design be detected.

The project began in the first week in September of 1982 and was completed in the second week of January, 1983.