6-2000

CFMMS – Computerized Facilities Maintenance Management System

Prawit Rotsawatsuk

CFMMS – COMPUTERIZED FACILITIES MAINTENANCE MANAGEMENT SYSTEM

by

Prawit Rotsawatsuk

A Thesis
Submitted to the
Faculty of The Graduate College
in partial fulfillment of the
requirements for the
Degree of Master of Science
Department of Construction Engineering, Materials Engineering,
and Industrial Design

Western Michigan University
Kalamazoo, Michigan
June 2000

ACKNOWLEDGEMENTS

I would like to express my gratitude to my thesis advisor, Dr. Mohammed E. Haque, P.E., for his encouragement, his valuable suggestions, his guidance, friendships, and especially his valuable time to this thesis. Also, I would like to thank the thesis committee members, Dr. Roman Rabiej and Mrs. Sai Ravichandran for their interest in this thesis.

I want to make special mention of my wife and my son, my closest collaborators, who endured the disruptions and emotional turmoil associated with working this thesis.

Finally, I wish to thank my parent for their patience, support, and encouragement while I am study and working on this thesis.

Prawit Rotsawatsuk

CFMMS – COMPUTERIZED FACILITIES MAINTENANCE MANAGEMENT SYSTEM

Prawit Rotsawatsuk, M.S.

Western Michigan University, 2000

Computerized facilities maintenance management systems, previously available only to very large facilities, are now accessible to small and medium-sized maintenance organizations. Advertisements for such products promise reductions in maintenance costs. Where a manual system of maintenance management is currently employed, moderate reductions in maintenance costs and facility downtime are fully possible. Where no formal maintenance management procedures are present, the structure and case of a computerized maintenance system can result in dramatic cost reductions and improvements in facility availability. There are, however, practical limits to the applicability of computerized maintenance systems. The objective of any maintenance program is to minimize the total costs resulting from the execution or lack of execution of proper facility maintenance. The computer, because of its ability to store and manipulate large amounts of data, can be a valuable asset to the facilities maintenance manager. A computer can quickly scan this data and report specific findings, trends, or discrepancies. The advantage of the computer over manual methods lies primarily in the ability to store, process, and report large volumes of various types of information.

TABLE OF CONTENTS

Table of Contents—continued

CHAPTER

Table of Contents—continued

CHAPTER

Table of Contents—continued

CHAPTER

Table of Contents—continued

LIST OF FIGURES

List of Figures—continued

CHAPTER I

INTRODUCTION

Facilities management is not a new or difficult concept to understand. By the age of four, most persons have had and comprehended the rudimentary experience of taking care of some possession, suffering when it functioned improperly, struggling to make it work, enjoying that possession when nothing was amiss, and paying a price when the possession was abused. Like a child with his favorite toy, the facility manager has the same fate and, all too typically, the same level of success and failure.

The difficulties inherent in the proper maintenance of a facility, whether it is a hospital, elementary school, industrial plant, office building or shopping mall, are compounded by the fact that it takes no specific level of education to identify when maintenance is working or when it has failed. To the facility manager, the world is full of critics; thanks are infrequent, perfection expected. The link between that perfection and profitability is tenuous, while the link between poor performance and losses is immediate. If the facility manager ever doubts that his or her job is important, he or she should try turning off the air conditioning for about an hour on a hot summer day.

The proper maintenance of a facility is easy to define and easy to evaluate. The importance of maintenance is easily established. Yet achieving an acceptable level of maintenance is extremely difficult. Generally, if everything is working properly, if cleanliness is up to standards, if the work force and clients are all happy, then one can expect that the facility manager be over budget. The facility manager therefore is a juggler; weighing the options of maintenance, repair, and replacement

against their respective costs; setting and resetting priorities; "putting out fires"; and, on a good day, feeling tremendous pride of accomplishment.

CHAPTER II

OBJECTIVES

The main objective of this thesis is to develop a computer-based facilities maintenance management system database program to easily store maintenance cost data and to easily access the database that will be replace the old paper-base facilities maintenance management.

Computerized facilities maintenance management previously available only to very large facilities, are now accessible to small and medium-sized maintenance organizations. Advertisements for such products promise reductions in maintenance costs. Such claims are well founded. Where a manual system of maintenance management is currently employed, moderate reductions in maintenance costs and facility downtime are fully possible. Where no formal maintenance management procedures are present, the structure and case of a computerized maintenance system can result in dramatic cost reductions and improvements in facility availability. There are, however, practical limits to the applicability of computerized maintenance systems. Very small organizations may not be able to justify the simplest and least expensive system. Intermediate-sized organizations may not be able to justify the cost of a very sophisticated computer system. Large maintenance organizations may find the effectiveness of their maintenance program encumbered by a system which has less than complete features. The task, then, is to match one of the many available computerized maintenance systems with the size and needs of a particular maintenance organization.

The objective of this facilities maintenance management program is to

minimize the total costs resulting from the execution or lack of execution of proper facility maintenance. Since these costs generally accrue in small increments through the execution of many small maintenance efforts, the ability to track each of these activities and their attendant costs is of great importance. Once such data is gathered, it must be interpreted and appropriate actions taken. The computer, because of its ability to store and manipulate large amounts of data, can be a valuable asset to the facilities maintenance manager. A computer can quickly scan this data and report specific findings, trends, or discrepancies. The advantage of the computer over manual methods lies primarily in the ability to store, process, and report large volumes of various types of information.

CHAPTER III

FACILITIES MAINTENANCE MANAGEMENT BACKGROUND

What is Facilities Maintenance Management?

Facilities maintenance management is not a new or difficult concept to understand. By the age of four, most persons have had and comprehended the rudimentary experience of taking care of some possession, suffering when it functioned improperly, struggling to make it work, enjoying that possession when nothing was amiss, and paying a price when the possession was abused. Like a child with his favorite toy, the facility manager has the same fate and, all too typically, the same level of success and failure.

The difficulties inherent in the proper maintenance of a facility, whether it is a hospital, elementary school, industrial plant, office building or shopping mall, are compounded by the fact that it takes no specific level of education to identify when maintenance is working or when it has failed. To the facility manager, the world is full of critics; thanks are infrequent, perfection expected. The link between that perfection and profitability is tenuous, while the link between poor performance and losses is immediate. If the facility manager ever doubts that his or her job is important, he or she should try turning off the air conditioning for about an hour on a hot summer day. Gregory, H.M. (1988)

The proper maintenance of a facility is easy to define and easy to evaluate. The importance of maintenance is easily established. Yet achieving an acceptable level of maintenance is extremely difficult. Generally, if everything is working

properly, if cleanliness is up to standards, if the work force and clients are all happy, then one can expect that the facility manager be over budget. The facility manager therefore is a juggler; weighing the options of maintenance, repair, and replacement against their respective costs; setting and resetting priorities; "putting out fires"; and, on a good day, feeling tremendous pride of accomplishment.

Life Cycle Stages of a Facility

The life of a facility, whether of one building or of an extensive complex, can be divided into several finite stages: design, construction, occupancy and use, repair, rehabilitation, and disposal. Similar stages apply for individual systems or components within a facility.

Within each stage there are additional definitive sets of actions required of the facility manager or owner and each of these actions has an attendant cost. For example, since all facilities or their component parts arise from some recognized need, it can be assumed that the facility (or component) is justified and must remain in place and in operation. Whether the need comes from an increase in population, initiating the construction of a new school, or is in response to a corporate objective to expand profits, causing construction of new office or plant facilities, the objective of providing that operating facility on schedule at the lowest possible cost is always paramount.

Prior to the birth of a new facility, all estimated potential costs are controllable. As soon as the life of the building or component begins, however, the control (or flexibility) of the total life cycle cost of the facility diminishes. Once the design for the facility is fixed, so too are the majority of the anticipated operating and maintenance costs. Few opportunities will arise in the life of the facility to markedly

change these operating, or life cycle, costs. Therefore, it is imperative to understand these maintenance and operating costs and to address them during the design stages. When a facility already exists, it is imperative that the maintenance effort be studied to uncover any possible reductions in operating and maintenance costs. These cost savings can compound for many years.

Once a facility or improvement to a facility is approved, the first step is to firmly define the need. With the specific needs in mind, the new facility is designed and construction begins. At the completion of construction, the facility is occupied and used. From this point forward the facility must be maintained in a usable state to assure that the original need is continually satisfied. When this need is no longer present, or the function obsolete, the facility or component is disposed of. Otherwise it will, at some point, require refurbishing or remodeling.

During the life of a facility the stages of design, construction, and disposal are brief periods when compared with the operation and maintenance stage. Initial construction, however, as the most costly stage, receives a "lion's share" of attention from the organization. Operation and maintenance is often accepted as a given, but cursed for its nagging presence. There is no glamour in maintenance, but, as a major long term cost, maintenance must be controlled. To be controlled, maintenance must be understood.

In the background of these readily identifiable, or direct, activities are numerous indirect functions which proceed simultaneously to ensure that the proper level of maintenance is sustained. These background activities include spare parts inventory and ordering, engineering support, accounting, payroll, and payment and billing for services received or rendered. Perhaps the most important indirect maintenance activity is the continuous comparison of budgeted maintenance costs to

actual expenditures cost accounting and control.

Direct Maintenance Work

Direct maintenance work includes activities, which preserve or restore the function of the facility. This category is divided into the following subdivisions: housekeeping, general maintenance, preventive maintenance, repair, replacement, improvement, modification and utilities. Every direct maintenance work item can be placed in one of these categories. This breakdown also defines, to a certain extent, the type of resources required to perform the work. The names of these categories are likely to vary slightly from one organization to another; but the basic elements, as described, must be present in every facility maintenance program. Gregory, H.M. (1988)

Housekeeping

Housekeeping is that group of activities which make the facility presentable and fully usable to its clients, preserving the proper day-to-day operation of a properly functioning facility. Included are basic cleaning of spaces, emptying of trash receptacles, replacement of towels and toilet paper, sweeping, mopping, and dusting. Housekeeping tasks are limited to those performed by unskilled labor on a frequent basis, generally daily.

General Maintenance

General maintenance might also be described as infrequent housekeeping. A general maintenance activity often requires somewhat more skill than housekeeping, and often uses specialized equipment. Typical general maintenance activities include

stripping and re-waxing floors, repainting walls and trim, spring planting flower beds, sweeping roads and parking lots, or steam cleaning carpets. General maintenance improves or preserves the appearance of a facility and is accomplished at discrete intervals based on seasonal considerations, accumulated experience, or aesthetic preferences. However, many general maintenance activities, if continually neglected, may lead to premature failure of the facility component.

Also included within, the category of general maintenance are nuisance work items such as replacing faucet washers, tightening loose door or window hardware, tightening valve gland packing, adjusting door closers, replacing light bulbs, and lubricating hinges.

Along with housekeeping, general maintenance is often the most visible of the various maintenance activities. The need for general maintenance is most frequently identified and reported by the facility user. General maintenance activities are rarely critical to operation of a facility. If ignored, however, more severe problems usually arise. Further, the users may cease reporting problems if no prompt action is ever taken to preserve this vital stream of user input.

Preventive Maintenance

Preventive maintenance is any work performed to an operational device or facility to continue operating at its proper efficiency without interruption. Preventive maintenance activities are performed at regular intervals, usually by a skilled work force. As an individual category, preventive maintenance is significantly different from general maintenance. The interval between preventive maintenance actions on a particular component is established by manufacturer's recommendations, empirical measurements of degrading performance, or the impending failure of an unmaintained

piece of equipment. When preventive maintenance is continually neglected, dramatic and costly failures often occur. For this reason, a formal preventive maintenance program should be a high priority. In the interest of cost, however, care must be taken not to allow a state of "over maintenance" to creep into the program. Once, the preventive maintenance program is formally in place, it must be continually examined to determine if the cost of frequent maintenance exceeds the cost of downtime and repair if no maintenance were performed. A suitable objective might be: Establish a program of routine inspection and services of equipment to prevent premature failures.

Repair

Repair work involves restoring to operation some component of the facility after it has failed. It is the "headline maker" of facilities maintenance, as failures rarely happen at convenient times. Most often the repairs must be made immediately, at the expense of other scheduled maintenance.

In establishing objectives for completing repairs, it is often necessary to set priorities based on the urgency of need for the repair. These priorities establish the desired maximum time required to complete repairs. Thereafter, as failures occur, the repair is classified by priority and work is scheduled accordingly. Work which is not immediately required (low priority) is often placed on a backlog for future scheduling and accomplishment.

Although the various priority groupings have finite desired time limits for accomplishment, the actual assignment of priorities to work is subjective. Therefore it is further necessary to establish criteria for each priority grouping. For example, this may seem redundant, assigning a priority to a project rather than setting a simple time

limit for completion, but it is necessary to direct the scheduling of work by shop personnel. Cost comparisons are often made at this stage when the magnitude of the failure indicates that replacement should be considered.

Replacement

Replacement, as a distinct work element, is confined to a program of planned replacement of facility components. It may be further limited to major components such as air conditioning compressors, furnaces or hot water heaters. Replacement is performed when the equipment has reached the end of its useful life, when it no longer can perform due to degradation of its internal components and repair is no longer cost effective. Included under the title of replacement would be the major rebuilding of any component, since rebuilding also restores performance.

Ideally, in a maintenance program with sufficient historical data on similar components and a means for noting component degeneration, the replacement of any facility component would be scheduled to occur just prior to total failure. Replacement is, therefore, the final step in an orderly maintenance program, which has extracted the most cost effective use out of a component.

Although the decision to replace a piece of equipment is generally inevitable, it is not without a wide variety of options. Accordingly, a program of planned replacement should revolve around the costs of the equipment, its installation, and its maintenance. The replacement program presents a unique opportunity to the facilities maintenance manager. Since component degeneration mandates some form of replacement, the manager can analyze the impact of using a different component that might result in a lower life cycle cost. In other words, an item that is initially more expensive might require less repair and less frequent replacement.

Improvement

Improvement projects enhance the proper operation or reduce the operating costs of a facility. These projects may include the installation of energy and utility conserving devices such as flow restricting faucets, thermal insulation, or more efficient heating or cooling system components. Replacement of properly operating but maintenance intensive equipment with similar but more reliable products is included in this category.

Life cycle cost is the essence of any improvement project. However, while any project, which provides for reduced long-term costs are worthwhile, improvement projects are generally costly and maintenance budgets may not be sufficient to sustain them.

While other maintenance activities are dictated by the facility or component in place, improvement projects are often initiated by maintenance management personnel. It is the responsibility of the maintenance staff to continually seek methods to reduce operation costs. A maintenance manager should seek to identify cost saving projects for consideration by upper management.

Modification

Modification projects alter the basic facility or facility component to accommodate a new function or corporate initiative. Modification projects differ from improvement projects by their point of origin. Modification projects are initiated from outside of the maintenance staff whereas improvement projects are initiated by maintenance staff. The estimated cost of a modification or alteration must be considered well in advance, since corporate decisions concerning new initiatives are often based on cost. Since the decision to go ahead with a modification project relies

heavily on the estimated costs.

Utilities

The direct work elements previously discussed have generally required the expenditure of on-site labor. Utilities are included as a direct work item since many facilities generate their own utilities. The utilities work element includes furnishing electrical power, potable water, centrally produced heat or cooling fluids, collection and disposal of sewage and other wastes, and collection and disposal of storm water.

Utilities are usually provided by local municipal utility systems for small facilities. In this case, the involvement of a facility or maintenance manager is minimal. For a small facility, the utility billings are simply verified and certified and the utility company called when service is interrupted and/or repairs is required.

In larger facilities, this may involve the full time operation of potable water wells and treatment plants, full electrical generating equipment, massive central boilers for distributed steam for heat or processing, and in many cases, the operation of a sewage treatment and disposal facility. Such systems become mini-facilities in themselves, requiring "round the clock" attention, and contribute substantially to the overall workload for maintenance and repair of the facility as a whole. For this reason, internally operated utility systems often represent a significant portion of the operating costs of a facility. Of course, the decision to internally provide these utilities was originally made because it was either cheaper, more reliable, or the service in question was not available in sufficient quantity from the municipal source.

Indirect Work Elements

Indirect work elements are the activities that facilitate the direct work

previously outlined. While not performed directly on a facility, these elements should be present in any cost effective program. Indirect work elements include: (a) work identification, (b) cost estimating, (c) purchasing, (d) supplies and inventory control, (e) scheduling, (f) work tracing and monitoring, (g) facility and equipment histories, and (h) engineering.

Work Identification

All physical work performed to a facility has a point of origin. Preventive maintenance, for example, is a planned activity whose frequency is either firmly established to prevent failure or is scheduled seasonally, such as switching over from heat to air-conditioning. Housekeeping is a regularly scheduled activity. General maintenance and repair work, however, are not scheduled and must first be noticed and then reported to the maintenance staff. In the course of preventive maintenance and formal daily rounds of equipment, any irregularities noted by the staff should be either corrected or logged for future repair. Costs prohibit continual maintenance staff inspection of all facilities and spaces, thus the facility user must be relied upon to identify and report repair and general maintenance discrepancies.

A non-functioning item is often first noted by the user. Yet, if the user is to be relied on as a source of identifying needed repairs, it is imperative that all users have an easy mechanism for reporting broken items. Without a simple method for contacting the maintenance staff, the user soon forgets to report the casually noticed items, assuming that noticing and reporting is someone else's responsibility. Since minor problems, when neglected, never cure them and invariably grow into major problems, it is essential that a readily accessible method for reporting problems be an integral part of a total facility maintenance program. A reporting system is most

typically established by use of a work receptionist who receives telephone notice of any discrepancy.

Major work performed by the maintenance staff under the categories of replacement, modification, or improvement is identified through various means. Replacement is predicted by the rate of decay in performance of the facility or equipment. Modifications are prescribed by the user to accommodate new or different facility functions. Improvements are the only projects that are voluntarily initiated. Although improvement projects are often facilitated by a simultaneous need for major replacements, the primary reasons for executing improvement projects are long-term monetary savings or increased profits.

A formal inspection program is another means for identifying work to be performed. This program might entail hourly checks of critical equipment or annual checks of unoccupied, inaccessible spaces in a building. The frequency of inspections would depend on the impact of failure of the inspected equipment or facility, should the early stages of degeneration go undetected. RSMeans (1996)

A formal maintenance objective for proper identification of work to be performed might read:

Conduct a program of formal inspection and facility monitoring scheduled to identify critically needed maintenance work, coupled with an open communication channel for non-staff members to report facility discrepancies.

Cost Estimating

Cost estimating is an integral part of all facets of facility maintenance. The role of cost estimating begins with the development of the annual operating and maintenance budget; it is necessary to predict the frequency and scope of the

maintenance work to be performed during the year. Historical records and trends are the most valuable tools available when preparing such a budget. From historical data the overall cost, or budget, for each area of work is developed. As individual repair work items are identified during the course of the year, it is necessary to produce a more refined estimate of the costs (to measure the progress of each individual task). A cost estimate may determine whether an item is repaired or replaced. Cost estimates play a similar role when evaluating the feasibility or desirability of modifications or improvements to the facility. Accurate costs depend on several variables. The prevailing wage rate, expected level of productivity, material costs, equipment costs, and overhead costs must be given full consideration. The relative predictability of these cost elements varies. Wage rates, for example, are usually very stable, while productivity levels are often unknown and highly dependent upon individual circumstances and work conditions. Thus, if the actual cause of the component failure is not immediately evident, the repair costs may be extremely difficult to estimate before the particular repair is completed. Since cost estimating is extremely difficult and often highly dependent on historical cost data and past experience, the task of preparing estimates is generally assigned to a limited number of persons within an organization.

Purchasing, Supplies, and Inventory Control

As maintenance work is carried out, there is a constant requirement for materials and supplies to sustain the work effort. This necessitates a formal system for predicting the materials required, procuring those materials, and maintaining accountability for those materials. The total material requirements and proper inventory levels for a budget period depends on two factors: the frequency of need for

the materials or parts in question, and the impact of not having a sufficient inventory of parts on hand on facility operations. Routine preventive maintenance tasks require a predictable type and quantity of materials. Repairs, being unpredictable, may cause interruptions to facility operations if appropriate parts are not readily available.

In addition to the procurement of materials and parts, it may be necessary to purchase services from outside repair specialists. Formal purchasing practices must be established to ensure that contractual obligations are established and met by servicing personnel. Where purchasing flexibility exists for selecting materials, suppliers, and service contractors from a source other than the lowest bidder, the maintenance manager must be prepared to assess the value of the product versus the price paid. The adage "you get what you pay for" is often very true in facility maintenance. For this reason, life cycle costs should be considered at each purchase or contract.

Cost Accounting and Control

Although most companies or organizations have formal accounting systems which concentrate on payables and receivables, it is necessary for the maintenance manager to have a parallel system to measure the costs of the various operations and transactions on a more frequent and more definitive basis. As computers gain a hold in an organization it is frequently possible to merge the two requirements. Lacking a suitable composite system, however, the manager may find it necessary to develop a cost management system to track costs by individual job, discrete facility functional area, direct work type, assigned work force or service contractor, and individual facility, equipment, or component. Additionally, the scope of the total maintenance effort may dictate subdivision of cost accounting to individual departments within the maintenance staff.

The purpose in cost accounting is to accurately measure the ongoing and historical costs of each maintenance activity. From this measurement, the maintenance manager can make decisions to redirect or reallocate resources to mitigate increasing costs or lessen stable costs. For example, excessive overtime costs might lead to hiring or reassigning additional full-time employees. Therefore, timeliness of reporting ongoing costs is an essential factor. The desired frequency of measuring and assessing ongoing costs varies depending on the risks associated with not measuring the costs. For example, preventive maintenance activities can be accurately estimated, are unlikely to vary significantly from the estimate, and need not be monitored frequently. Repair project costs, however, may fluctuate hourly and require frequent monitoring. The potential for costs to stray from the budgeted amount is limited for fixed price service contracts but can be great for undefined "open, inspect, and repair" work. Since the maintenance manager's time is always limited, efforts should be concentrated on the highest risks. Since the possibility for long-term over-budget problems can develop by a compilation of slight overages the manager must also have a method for tracking cumulative costs within the various categories mentioned above.

Scheduling

Since maintenance work elements are both planned (preventive maintenance, Improvement, modifications, housekeeping) and unplanned (repairs), the maintenance manager must be an efficient scheduler. It is necessary to provide sufficient scheduled workers for the jobs planned while maintaining sufficient flexibility to handle most unforeseen events. To adequately plan for anticipated and unanticipated work items, two elements must be scheduled: the work and the workers. As noted earlier,

preventive maintenance work is scheduled at defined intervals based upon need and risk. These work items must, therefore, be set down as fixed work and staff provided for their accomplishment. Similarly, known or surprise repair work must be covered. Once identified, individual workers must be assigned specific tasks for finite periods. This scheduling is based on the accurate productivity estimates established for each of the tasks.

From the known workload of planned tasks and the historically predicted incidence of repair work, an optimum staffing level (based on historical data) is established. Ideally, all planned work is accomplished and the backlog of repair work does not grow, if the staffing is adequate. Generally it is better to be slightly understaffed, resorting to overtime, than to be overstaffed which may lead to decreases in productivity as workers slow down to fill the day.

Work Tracking and Monitoring

In addition to monitoring the costs of the maintenance effort, attention must also be paid to ensuring that all work is completed in a timely manner. Since deferred preventive maintenance or neglected repair work can frequently lead to costly repairs or facility downtime, a system for tracking the identification and accomplishment of all work is imperative. If client-identified minor repairs are not accomplished in a reasonable amount of time, the client or user may soon decide it is useless to report minor problems. If minor problems are not reported, they could lead to major problems, therefore, the work tracking system must be timely and complete. The method for tracking work should establish defined intervals for workers or foremen to report completion. The length of the interval depends on the anticipated effect of not completing the work. For example, a weekly report of all routine work may be

sufficient for certain repairs, while hourly reports for emergency repairs may be too far apart. A primary reason for having a work tracking system is simply to ensure that no required work is forgotten.

The work tracking system should be logically tied to both the cost accounting and work scheduling systems. It must have the capability to track ongoing work progress (and costs) that exceeds regular reporting intervals. The system should also be able to provide feedback to both the facilities maintenance staff and the facility users concerning work status.

Facility and Equipment Histories

In order to properly predict and adjust the maintenance program for a facility, the manager must have an intimate knowledge of the life of the particular facility and, therefore, of facility components. Only with a total building history can proper decisions be made regarding the best course of maintenance actions for a given problem. Accurate recordings of actual construction, all modifications, and improvements should be kept. Major and recurrent minor repairs must be tracked. Periodic inspections must be made and the current state of the facility assessed and recorded. The same types of records should be kept for major equipment components. The purpose of accurate records of the total facility is to enable immediate troubleshooting and repairs. Shearer, K.A. (1922)

A proper equipment history should record the make and model of the equipment, date of installation, all major repairs, all preventive maintenance, routine parts replacements, and any continuing engineering measurements of equipment performance. Additionally, where the work force is stable, the name of the worker performing the repair or preventive maintenance should be noted. The equipment

history is then used to predict the need for eventual replacement or rebuilding of the component, or when trying to diagnose an unexpected failure that can often be traced to a recent repair.

Engineering

The previous direct maintenance work elements and associated indirect work can usually be accommodated without a formal in-house engineering staff. Many work elements, however, require engineering input at some stage. Engineering services include assessing facility and component performance, designing modifications to facilities or equipment, preparing engineering drawings and specifications for repairs, and troubleshooting major component or system failures.

Many large facilities may operate for long periods of time without formal engineering assistance. When, however, the facility manager needs technical assistance it is often expensive since the engineer must gain familiarity with the facility. For this reason, it is desirable to maintain an engineer or engineering firm on continual retainer, requiring that firm to stay conversant with the facility functions and physical components. The engineering disciplines predominately involved in facility maintenance are mechanical engineering, civil engineering, architectural engineering, and electrical engineering.

The need for engineering input is usually proportional to the size and complexity of the facility. If a facility were to grow in an orderly fashion, adding buildings and components, there might be a time at which the services of an engineering consultant become so frequent and the costs so great that in-house engineering capability may be more practical and efficient. In addition to pure cost, the need for more immediate analysis and response than that which outside

engineering services can render may dictate the decision to expand the maintenance staff with more engineering talent. There is no set facility size that dictates in-house engineering capability; the need varies based on facility function as well.

CHAPTER IV

COMPUTERIZED FACILITIES MAINTENANCE MANAGEMENT

Reasons for Computerization

The objective of any maintenance program is to minimize the total costs resulting from the execution or lack of execution of proper facility maintenance. Since these costs generally accrue in small increments through the execution of many small maintenance efforts, the ability to track each of these activities and their attendant costs is of great importance. Once such data is gathered, it must be interpreted and appropriate actions taken. The computer, because of its ability to store and manipulate large amounts of data, can be a valuable asset to the facilities maintenance manager. A computer can quickly scan this data and report specific findings, trends, or discrepancies. The advantage of the computer over manual methods lies primarily in the ability to store, process, and report large volumes of various types of information.

Information Storage and Processing

For each individual maintenance activity, at least two pieces of information are necessary: what needs to be done and when it must be done. If resources to perform maintenance were not limited and if the facility could accommodate interruption at any time the work list could be kept manually, sorted by "due date" and executed in order. Obviously maintenance is not that simple. Resources are limited, cost is important, and the maintenance effort must accommodate continued operation of the facility. When these factors, the volume of data related to each activity

increases proportionally. Other factors include the location of the project and the shop responsible for the project. Each of these facts related to an individual project can be listed on a single sheet of paper and stored until the project is to be executed. The total maintenance backlog would then be a single pile of individual projects. RSMeans (1996)

The different information related to an individual project has numerous applications. A list of all projects for a single shop is needed. A listing of all projects for a single building or facility function are also important. A summary of past costs sorted by building, shop, or type of work would be useful to analyze trends. This data can be used to organize and manage the maintenance effort. However, retrieval of this information requires a considerable amount of manual searching. A computer can save time during information storage, retrieval, sorting and reporting. Once salient data concerning an individual project is stored within the computer, it can be manipulated and combined with other project data in endless combinations, saving time otherwise spent in manual compilation of data. Modem data storage devices, such as hard disks, can retain millions of bytes of data.

Automated Calculations

Most maintenance management actions follow a logical pattern; each project follows a series of common steps. A project is identified, evaluated, given a priority, scheduled for accomplishment, assigned to a craftsman or service contractor, completed, and inspected. Data from the project is recorded for historical records, including the estimated and actual costs of materials, equipment, and labor. A computer program may duplicate many of these steps, and the speed of calculation and data processing of a computer produces immediate results. Manual calculations

are more time-consuming. There are many maintenance activities that are regularly scheduled, but only at great time Intervals; the computer provides automatic recall of these requirements, reducing the possibility that such maintenance activities will be forgotten. However, the move from manual record keeping to computerized maintenance management is a costly one. In addition to the purchase cost of the computers and computer programs, the move to a computer system implies a long-term commitment of labor to enter and retrieve the data from the computer. Those costs must be compared to the anticipated benefits of a computer system before this major commitment is made.

Standard Program Features

The maintenance procedures represent the logical evolution of maintenance as a managed technical activity and are employed with minor variations in most well run maintenance organizations. Because of this commonality of approach within the industry, computer programs, or software, have been developed which allow for immediate computerization of a maintenance program. The following sections briefly describe the basic factors of each standard feature. Following each description is a summary of the available enhancements to each feature. Not all commercially available programs contain every one of these enhancements; therefore, the facilities maintenance manager should examine each of these features and note the attendant enhancements. A listing of the essential features desired should be used as a checklist for the evaluation of the various commercial software packages. RSMeans (1996)

Where applicable, computer printed forms and reports are shown. Most maintenance management software packages produce varying versions of these forms and reports.

Unplanned Work Processing

All maintenance programs must react to unforeseen maintenance activities. The scope of the work and its location must be captured. As the project is evaluated, planned, scheduled, and assigned, more information is obtained. When the work is in progress and eventually completed, the amount of information grows. This information must be entered, transmitted to appropriate personnel, continually updated, and compiled to produce weekly, monthly, or annual reports. Gregory, H.M. (1988)

Entry

All initial data must be entered manually into the computer. This is generally performed by the person manning the work reception desk as a phoned in request is made or as time permits for those requests submitted in writing. The specific form used for requesting work should be identical to that produced by the computer for phoned in requests, as the work request should be a constant thread throughout the processing of maintenance work. The work request is used for describing the work to the worker, who records on it the maintenance results.

There are few variations for this simple step. In very sophisticated systems, the facility user's computer terminals are linked to the maintenance computer, allowing for direct user entry of data.

Transmitting

The work request becomes a work order during the evaluation. Once the work order has been approved for execution, it is transmitted to the shop foreman, or for smaller organizations, directly to the maintenance craftsman. Usually this transmittal

is made manually by dispatching the hard copy original work request to the shop or worker. If a computer is utilized, a computer-generated work order is sent to the worker in place of the original work request.

The transmittal of work orders can be enhanced by electronic transmission of the data. This is done through remote printer, located in each of the shop areas. The work order is dispatched electronically by the trouble desk after approval, estimates, and priority are determined. The exact copy of the work order is printed out in the appropriate location, eliminating the need for manual handling. This feature is particularly helpful where maintenance control and maintenance execution staffs are geographically separated. For emergency work orders, it is necessary however, to ensure that the work order has been received at the remote site. Oral communication by phone, radio, or beeper is usually desirable

Updating

Once maintenance work has been performed, the results of that effort are directly recorded on the work order by the maintenance worker. This data is reviewed and verified by the shop foreman and forwarded to maintenance control for updating of the original work order entry. When this data is input into a computer, records are updated and analyzed.

Few computer systems handle this task in any refined manner. It is possible for the maintenance shop to directly update the work order data at a remote terminal. However, only a very large organization will benefit from this feature. Usually the volume of data to be entered can be handled by the work reception/trouble desk clerk during his or her spare time.

Reporting

The results of an individual work order are important to several parties. The shop foreman is concerned with the total listing of incomplete work orders for each particular shop. The facility user is concerned with the status of work requests for their respective spaces or buildings. The maintenance manager is concerned with the timeliness of response to individual work orders and the relative workload of each shop. These multiple demands are handled by repeating the data from the total listing of work orders in different formats, sorted into different categories, and listed in different orders.

The flexibility of any computerized maintenance management system being considered is most evident in this area. A printout of subsets of the total backlog of work order, should be available for the following categories: (a) all work order, for each shop; (b) all work orders for each building or facility area; (c) all work orders due for completion by a specific time; (d) all work orders completed by a specific craftsman; (e) all work orders of a common type -repair, replacement, modification, etc; (f) work orders by differing dollar value and man-hour requirements; (g) work orders to, common equipment types; (h) all completed work orders for the above categories; (i) all pending work orders for the above categories; (j) all work orders which are overdue for the entire facility; and (k) all work orders which are overdue for a particular shop.

It is also desirable to sort each of the above reports in order by some criteria Common sorting options include: (a) by due date, (b) by shop, (c) by craftsman, (d) by priority, (e) by dollar value, (f) by date of request, and (g) by work type.

In addition to the standard reports listed above, it is desirable to have an ad *hoc query* capability for report generation, This feature, common to many database

systems, allows for the generation of most any custom report needed by the maintenance management staff. A sample report, which does not fit the standard categories listed above, might be:

A listing of all incomplete repair work orders of priority 2 or higher which are assigned to shops A, B & D.

The ability of computerized systems to handle this type of request varies considerably. In some sophisticated systems, this request may be handled by simple English commands typed in by the maintenance manager. In less sophisticated systems, such a report would require the direct programming of the computer by a trained computer programmer. Since the cost of a computer software system increases dramatically with increases in flexibility, the prospective purchaser should consider the possible report options carefully and determine the minimum requirements for the specific maintenance program.

Planned Work Processing

Planned work occurs at regular intervals and has a predictable cost and duration. Preventive maintenance orders and other recurring jobs are the most common categories of planned work. Once a specific task is recorded in the computer and it, frequency established, the computer recalls that data and automatically generates a requirement for its timely completion. Ensuring that this necessary work is not forgotten is the primary asset of a computerized maintenance management system. Planned work must be scheduled, monitored, general progress reports made, and supplies ordered. A computer can greatly enhance these functions, as described in the following section. Gregory, H.M. (1988)

Recording

Once identified by the maintenance staff, planned work order, are entered into the computer manually. A typical planned work order, or PMO, or preventive Maintenance Order (PMO), is entered by the work receptionist/trouble desk clerk. Various systems provide varying PMO formats, but initial entry of all PMO's and recurring work orders is a one-time, labor-intensive task. The planned work order may be entered by the work reception/trouble desk clerk on a time available basis or, if necessary, contracted to temporary help.

A few computerized maintenance management systems have pre-programmed Preventive Maintenance Orders that can he customized or adopted as is. This feature can save a considerable amount of data entry time. It may also call attention to common PMO's which might not have been identified for the particular facility and equipment being maintained.

Scheduling

Computer scheduling involves the automatic recall of individual work items from a memory. The work items, usually Preventive Maintenance Orders, are printed when they are due for completion. Simple programs search the backlog of PMO's and print those due for work, not taking into account the available staff to complete the work. In such cases, the workload is balanced from one period to the next by input of the due date and subsequent frequency at the time of original data input.

The output can lie made in several forms. The simplest is a listing of PMO's to be done. Armed with the listing, the shop foreman pulls individual PMO instructions from hard copy files and assigns them to a worker. In this system only the title of the PMO, some identifying number, the due date and frequency are stored in the

computer.

The scheduling provisions of a maintenance compute, program can provide great assistance to the maintenance manager and shop foremen in balancing the workload for their work force. Since there is a varying level of unplanned work that has to be accommodated, the amount of planned work which should he scheduled each week will vary. The computer program that simply prints the listing of all work needed places the burden for balancing this workload on the foreman. More sophisticated programs will perform this balancing function by analyzing the total workload and printing a schedule (of planned and unplanned work) which fully utilizes the available staff. These programs combine the current unplanned and planned work items, in order of which should be done first. These choices are made by the computer using input management standards considering the overall maintenance goals and the individually priorities for planned and unplanned work orders.

Monitoring

Once planned work has been assigned, it should be tracked through to completion. Computer records are updated to reflect the time of completion and the man-hours and materials expended. Most computer systems have the capability to print a listing of all planned work that was not reported as completed, or deferred maintenance items.

In addition to compiling a record of completed and incomplete work, the computer may be able to update individual equipment histories with the results of each PMO, depending on the sophistication of the computer and the accompanying software. Since additional needed work many be uncovered in the course of

completing a PMO, a provision in the computer system might also allow for automatic generation of a new unplanned work order.

Reporting

The planned work reporting function of computerized management system are similar to those previously described for unplanned work reporting. Listings of planned work by the numerous categories, sorted by the various attributes are desirable. In addition to these reports, a single report of all PMO's for each particular piece of equipment should be provided. As mentioned with unplanned work reporting, the feature if ad hoc query is very desirable for reporting of planned work.

Ordering

Since each planned work activity requires known types and quantities of materials and supplies for accomplishment, the order of these supplies can be tracked by the computer based on the scheduling of the PMO or recurring work activity. The computer may compile a list of required materials for individual or collective work activities for defined periods. Such a listing could be made for all known material and supply requirements for the next three periods. This listing ensures that sufficient materials are ordered well in advance.

The capability to identify material requirements may be absent from many computerized systems. Sophistication in this area may also include an automatic updating of estimated inventory levels of key supplies, materials, and spare parts. This involves internal computer file linkage between the planned work and inventory control dam.

Facility Histories

As individual planned and unplanned work activities are completed, the data regarding the date of completion, man-hours expended, material, supplies, and spare pact expenditures are recorded. This data forms a maintenance history for the facility, the individual building, and the specific equipment upon which the work was performed. Computer enhancements of these functions are described in the following sections.

Building Data

The long-term observation of maintenance costs by facility management can significantly enhance the forecasting of future maintenance costs. While planned work is definitive in scope and frequency, unplanned work is not; as a facility ages, the cost of maintenance is likely to increase. Facility discrepancies requiring maintenance action may increase in frequency, but in a subtle nature that is not readily observed. Records kept by the computer, tracing individual actions to each building or large component space within a building allow the manager, to observe trends indicating such an increase in maintenance costs. The extent to which these trends are traceable depends on the level of definition of the location of the individual work activities. In order to ensure maximum tractability, specific locations of the work must be defined by building, floor, and room.

Since the above-mentioned data is easily captured, most systems have some format for reporting total building costs. Enhancements in this area include automatic calculation of costs per square foot, costs per maintenance visit, cost by various maintenance activity type, etc. This data may also be described graphically, providing visual evidence of any trends.

Equipment Data

Most computer systems are capable of storing the maintenance history of specific pieces of equipment. These equipment histories are valuable tools in predicting times for replacement and for tracing possible maintenance-related causes for equipment failure. With such specific records, the maintenance manager is well informed when deciding whether to repair or replace a failed piece of equipment. The equipment history is a chronological compilation of the maintenance activities of initial installation, Preventive maintenance, major and minor repairs, rebuilding, and eventual replacement. Specific pans replaced are cited directly or implied by the notation of a completed Preventive Maintenance Order that contains routine replacement of parts.

The ability of the computer to consolidate and report maintenance activities by specific pieces of equipment is a function of how well individual work activities are recorded. Preventive maintenance order records automatically link work to equipment, but unplanned work must be noted separately on work orders. Enhanced systems link these events together automatically, eliminating the need for dual entry of data.

Maintenance Cost Accounting

Cost is a continual consideration, since manpower utilization, material consumption, and facility down time all require expenditure of company resources or represent lost profits. In order to track specific facility and equipment maintenance costs, records of maintenance activities are kept by individual activity every day. Costs accrue, however, on a less basis. Payrolls are made at weekly or bi-weekly intervals. Materials supplies are purchased in bulk. Spare pans are purchased

infrequently.

Due to the difference in the time that resources are expended and when they are paid for, the maintenance management computer system may not be able to make a direct comparison between individual activity cast and actual dollar expenditure. Such a linkage is not mandatory. The system should, however, be able to report costs that approximately total the actual expenditure of fund, through the payment of payrolls and material invoices. Accurate dollar costs of individual activities are only necessary when such work is to be reimbursed.

To consolidate two separate sets of records (one for maintenance activity and one for maintenance cost,) to one cost accounting system is a major evolution with few virtues to justify the tremendous difficulties involved. Exact financial accounting deal with finite instruments such as timecards, purchase orders, and invoices. Control of these items can be left with a few individuals, allowing for greater accuracy of records. If the costs were linked and accounted for exactly at the individual maintenance activity level the number of people involved, and hence the chance for error increase considerably. For these reasons most computerized maintenance management systems track maintenance costs in parallel with a formal committing system, but do not usually directly link and reconcile the two systems.

Labor Utilization

Labor is tracked by the number of man-hours expended upon an individual maintenance activity. The total of all man-hours expended on activities during a week should equal the total number of man-hours available and the number of man-hours paid for in payroll. Attempts are made to reconcile any differences. The actual reported hours are retrievable in report, of completed Work, by categories of shop,

type of work, etc. Computerized so stems generally allow the name or code of the worker to be recorded within the computer records for each maintenance activity. This proves helpful when tracing personnel-related maintenance problems.

The degree of accuracy of the total man-hour cost varies with each system. Sophisticated systems require input of actual employee names. The computer then links the number of hours worked to each employee's rate of pay to determine an actual, exact cost for the labor expended. The degree of accuracy is not always necessary. These sophisticated systems are most useful when large amounts of maintenance work are conducted on a reimbursable basis.

Equipment and Tool Costs

The cost of using normal shop tools is not tracked to individual maintenance activates. Computer systems tabulate some tool costs in the cost of completing a work order, usually when tools are specifically rented or purchased for one time use.

Material Costs

Material costs include consumable supplies, raw construction materials, and specific spare pans. Consumable supplies ate not usually charged to individual work orders unless used in major quantity or used exclusively for this one work order. If consumable are drawn from stock, which was purchased in bulk, the cost can be pro-rated. However, such costs are usually included in overhead costs.

The estimated cost of raw construction materials is charged to specific work orders. Although care is taken to estimate the most accurate cost, for raw materials, exact inventory control and costing is riot usually worth the time necessary to determine the exact price, and quantities. Spare parts, however, specifically required

by a single price of equipment are charged to that equipment through exact costing on all work order

Tracing material costs to a work order can be done with great accuracy Enhanced computer systems provide direct linkage between individual work orders and the inventory control system to produce absolute material costs.

Inventory Control

The key to prompt maintenance response is the availability of sufficient to perform the needed work. Additional personnel can be hired or diverted from other tasks. Tools can be rented. Materials, whether they are consumable supplies or specialized spare parts, however, are not always readily available. For this reason, once the type and quantity of materials to be maintained on hand is defined, a system must be established to determine the availability and location of these materials, without extensive search. Most computer systems have provisions for tracking materials through a formal inventory control system. In its simplest form, such capability is limited to an independent listing of item name or part number, and the respective quantity on hand.

Since inventory control is the tracking of the exact number of pint, in stock, the sophisticated system will coordinate purchases with increases to the computer inventory levels. As materials are consumed, the reported consumption is noted and reflected in the resulting inventory levels. For rapidly consumed materials, some inventory systems may identify low levels and automatically generate orders for the proper quantity. If desired, the computer may be able to use actual and exact material costs to compute the final reported cost of work order. For such a system the worker simply cites the material type and quantity and the attendant cost is extracted from the

computer inventory records.

Any computerized inventory control system is accurate only if regularly verified by comparison of the computer records with actual on-hand stock. The frequency of this comparison is determined by the facilities maintenance manager, based on actual consumption.

Maintenance Management Reports

The maintenance management staff usually desires various reports to measure the overall effectiveness of the maintenance program. These reports will indicate any trends in changing maintenance costs or types of maintenance.

Management reporting capability is of great importance, since all data is gathered with the intent of enhancing management functions and reducing maintenance costs. This area is also subject to varying needs by varying managers with different management styles and intentions. Again, the *ad-hoc query* feature is most important, allowing for the generation of reports of all types.

General System Features

Previous sections have described the basic structure and function, of a computerized maintenance management system. Both common and desirable features have been outlined. Without attempting to define and explain the nuances of computer programming arid database design there are additional, and more general, features of computer program, that merit strong consideration when selecting a system.

Data Integrity

Computers have no inherent intelligence. When a person sees the Carpenter Shop described as "Carp Shop," "Carpentry Shop," or even misspelled as "Carpenter Shop," it is obvious that all of these mean the same thing. The computer, however, recognizes these as four different shops. Since data are grouped by many different categories based on entries such as shop name, building name, or work type all such entries should be consistent. Extra care taken by the data entry Clerk can eliminate many, but not all errors. More sophisticated computer programs have crosschecking features where data are double-checked the time of entry against specifically allowed data entries. If, for extra the name of a shop does not match one in a previously defined list of shop names, the clerk is alerted that the entry is invalid and must be corrected. Individual entries may also be automatically rejected by the computer if the entry does not match a pattern (such as all numbers or all letters) or if the entry is not within a range of allowed values. This feature reduces the chance that records might be entered or classified incorrectly, misinterpreted, or lost.

Archiving Data

When a computerized maintenance system is purchased, significant investments of time are necessary to enter and update data for the system. Previously kept manual records are disposed of if they duplicate the computer-stored records. The computer data are stored on magnetic media, on hard or floppy disks. This media is volatile and subject to unexpected failure, denying access to some or all records. While the failures are rare events, they may be catastrophic unless protection is provided to recover past work. This may be prevented if daily back-ups are made of each file.

Regular Back-ups

Computer hardware and software must provide a system for backing up stored data. This should be done on an external copy to protect the information should a computer failure occur. These backups should be made daily, or as often as significant changes are made to the stored data. The entire disk should be backed no once every two weeks. Thus, if a failure occurs the documents can be copied in full (except the day's work) from the back-up tapes or disks. The remainder of the work is then reentered.

Long-Term Archived Files

As the computer system is laid, considerable data accumulates in the permanent memory. As time passes the need to access this old data is reduced and eventually becomes nonexistent. When such data is no longer used regularly, it should be archived to a floppy disk or tape backup and deleted from the hard disk. This saves room on the disk and speeds the bi-weekly back-up proms. Although the need for accessing this old data may seem nonexistent, the cost of the archive disks or tape is very small. If the need does arise the data can be reloaded into the computer and accessed cry quickly.

Hardware Considerations

All computers do not operate with all software; most software is written to run on specific computers or classes of computers. In most cases, an exact match is necessary. Further, maintenance management software is but one of many applications for which a computer is helpful within the daily operation of a maintenance organization.

Establishing Objectives

The maintenance manager should formally describe the expected gains from the implementation of a computerized maintenance management system. These objectives may include goals such as the following: (a) reduced downtime due to poor maintenance, (b) reduced maintenance costs, and (c) reduced permanent maintenance staffing.

Identifying and Evaluating Alternatives

The various types of software available can be identified through search of trade literature or in consultation with various computer suppliers. If then, is my doubt as to whether particular software will meet the organizational needs and objectives, sales brochures or demonstration packages should be requested before engaging in a full demonstration.

The maintenance manager should construct a list of required features. This can be done using the features listed in earlier sections of this chapter. Features which are desirable but not required should also be listed. Each possible product should be compared to this list.

The list should be prioritized to identify those essential, desired features and enhancements that are important for the maintenance program. If product does not provide these minimum requirements, it should be eliminated from further consideration. Only after the field of choices is narrowed should the cost of the system be considered. The final decision should be cost-based. If the cost of purchasing and converting to a new system is unlikely to produce justifying saving, in reduced maintenance, the status quo should be maintained.

Costs should be recorded on cost summary form. One form should be

completed for each alternative that meets the minimum requirements. It should be noted that computer software is an area where least cost should not be the absolute determining factor. Ease of use, availability of training, and degree of after installation support should be given equal consideration.

Implementing the System

Regardless of the advertised advantages of the selected system, little success in the system's use results without firm management commitment to make the system work. That commitment must be a combination of patience, perseverance, and understanding. Difficulties should be expected, and system discipline maintained. Worker feedback is also important. These issues are addressed in the following sections.

Expected Difficulties

There are always some difficulties encountered when adopting a new system. The personnel most involved with the new computer system may be frustrated by the interaction with the computer, or not comfortable with the order in which information is requested or printed. Craftsmen may perceive some evil purpose behind the system, such as it being a means of checking their productivity. Certain individuals may be scared that they might somehow damage the computer program and software and refuse to touch the keyboard. A typical pattern of progressive feelings toward the system might be the following: (a) stark fear, (b) lingering apprehension, (c) begrudging acceptance, (d) tacit acceptance, (e) enthusiastic acceptance, (f) desire for system sophistication or improvement, and (g) wonderment at how the organization ever managed without the computer.

Each employee who interacts with the computer in any way will welcome the system at one of these progressive feeling levels. It is management's role to ensure that they move forward up the chain of acceptance. This is often best handled by example. One enthusiastic individual from my level can inspire similar attempts at acceptance. In addition to problems with personnel accepting the system, there are often some minor problems with the computer hardware or soma re. These are normal but frustrating. They can be minimized by requiring full operation and checkout of the system by the supplying vendor(s). Where computers and software are purchased from separate parties disputes between these suppliers as to probable causes of problems will inevitably arise. These problems can be very damaging to operator acceptance unless they are anticipated as normal events.

Summary of Computerized Facilities Maintenance Management

System is a major investment. A need, such as a desire to reduce maintenance costs or improve facility and equipment reliability, is present. Computers offer certain capabilities that can satisfy those needs. Computer programs are readily available and somewhat similar in basic structure. Sufficient differences between programs exist to satisfy a wide variety of maintenance management needs. If management needs are well defined, alternative programs well evaluated, and finally, if the expected benefits exceed the estimated costs, the selected computer system should result in an improved maintenance program.

CHAPTER V

PROGRAM ARCHITECTURE AND DEVELOPMENT METHODOLOGY

The Architecture of CFMMS

The architecture of CFMMS is mainly a database system. The main function of database is use to store data. The database that has data can be much useful in either easily data input or easily data usage.

Database

CFMMS database is consisting of data show in Figure 1.

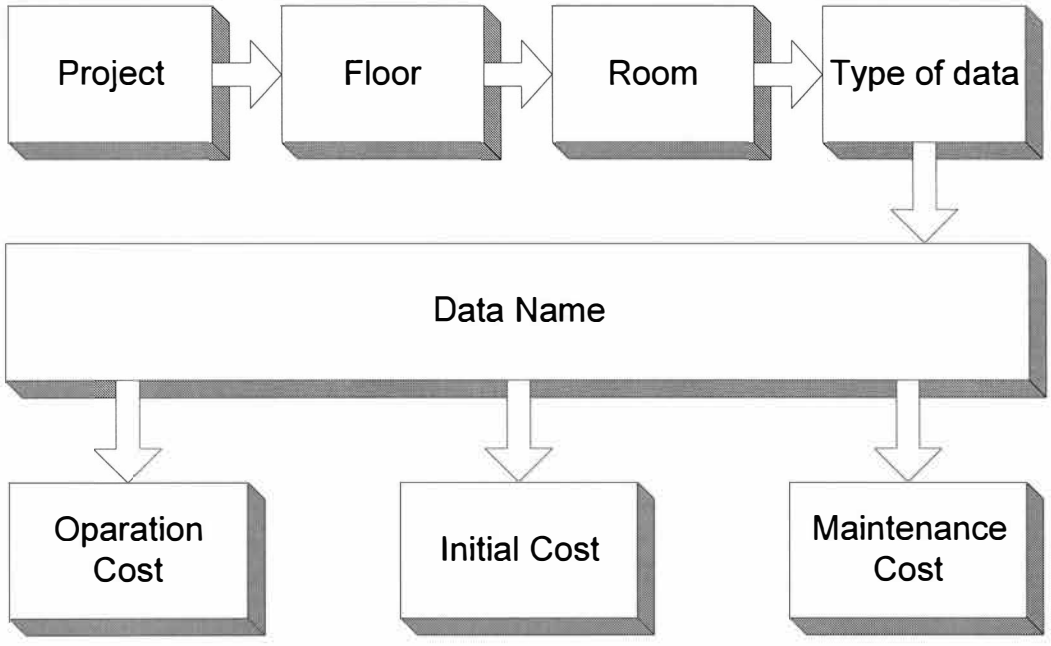

Figure 1. Database Table Flow Chart Relationships.

Project

Project is a table of database consists of project number and project name. The project can be either project name in case of that project has only one building or building name in case of that project has more than one building. Project number is used for link with floor. In several building project, either way, project name is a name of building. The project structure is shown in Figure 2.

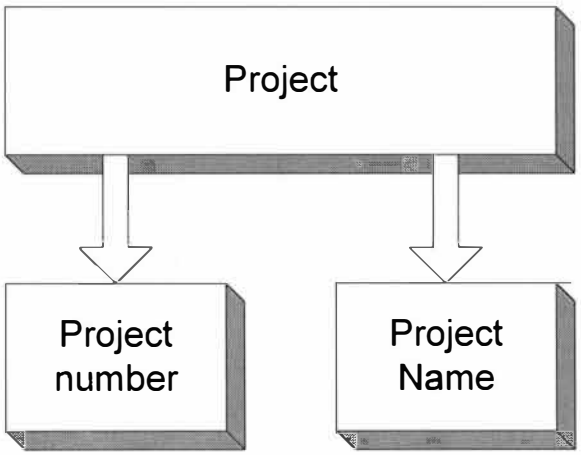

Figure 2. Project Table Consists of Project ID or Project Number and Project Name.

Floor

Floor is a table of floor consists of floor number, floor name, and project number. Floor number is the same function as project number. Floor name is the name of floor. Project number in floor table is used to identify the floor in the building. The project number in floor is identical to the project number in project table. It is useful in case of project that has more than one building. The floor table structure is shown in Figure 3.

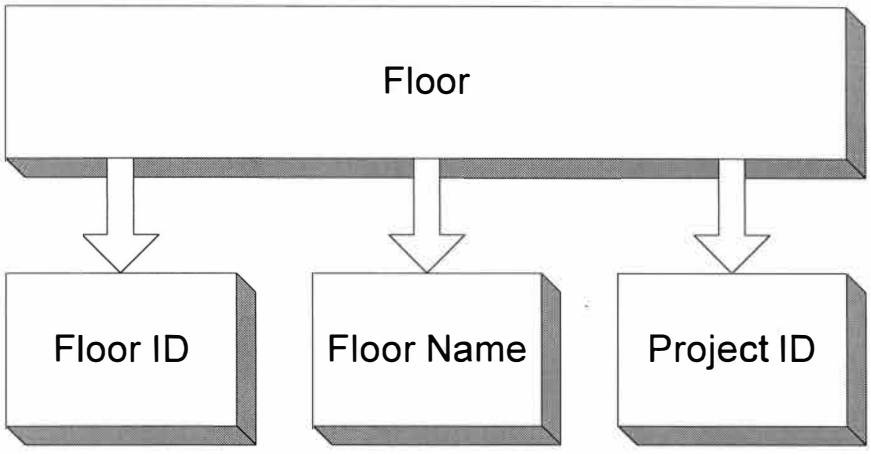

Figure 3. Floor Table Consists of Floor ID, Floor Name, and Project ID.

Room

Room table is consists of room number, room name, and floor number. Room number, room name, and floor number have the same function as floor number, floor name, and project number in floor table. The room structure is show in Figure 4.

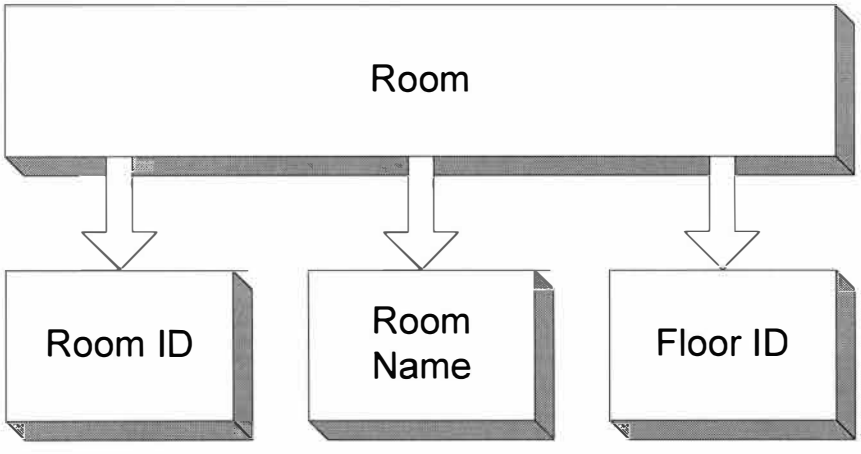

Figure 4. Room Table Consists of Room ID, Room Name, and Floor ID.

Data Type

Data type table is consist data type number, data type name, and room number. All data type fields have same function as of room table, and floor table. The table structure of data type is shown in Figure 5.

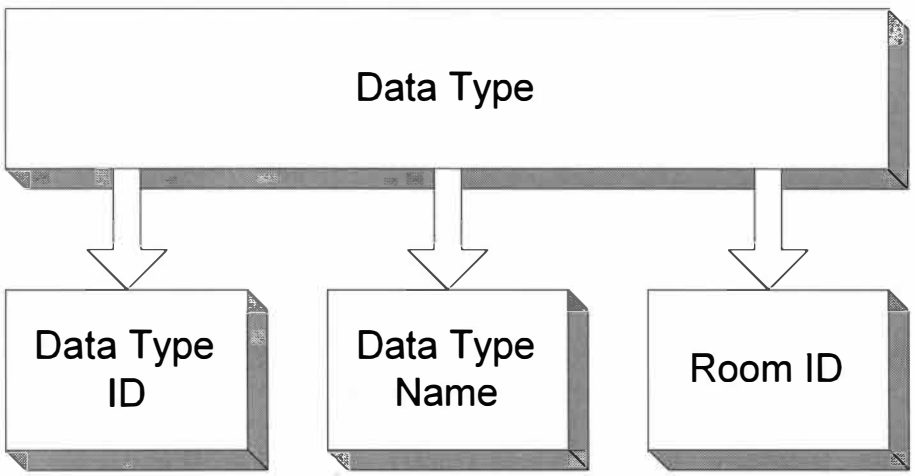

Figure 5. Data Type Table Consists of Data Type ID, Data Type Name, and Room ID.

Item Name

Item name table is consists of item-name number, item-name name, and data-type number. All of these fields have the same function as of floor, room, and data-type. The item name is the table that store data of every item in the project. Each item has three important cost data that are management cost, operating cost, and maintenance cost as shown in Figure 6.

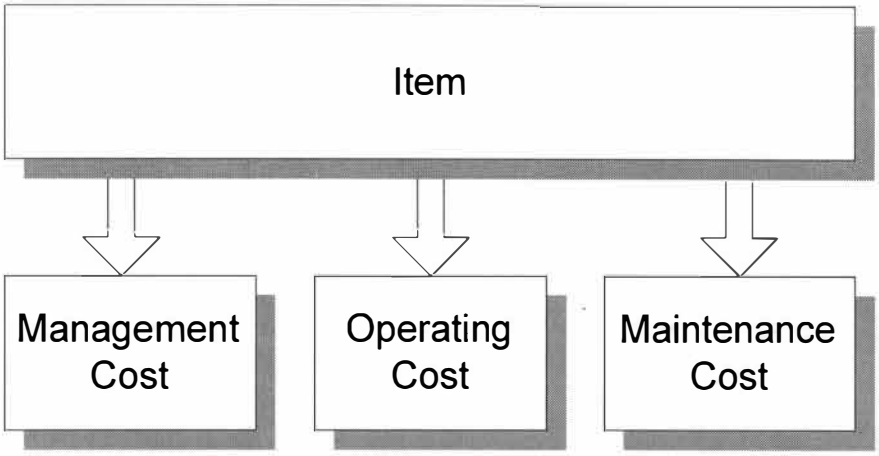

Figure 6. Item Number Consists of Management Cost, Operating Cost and Maintenance Cost.

The table structure of data type is shown in Figure 7.

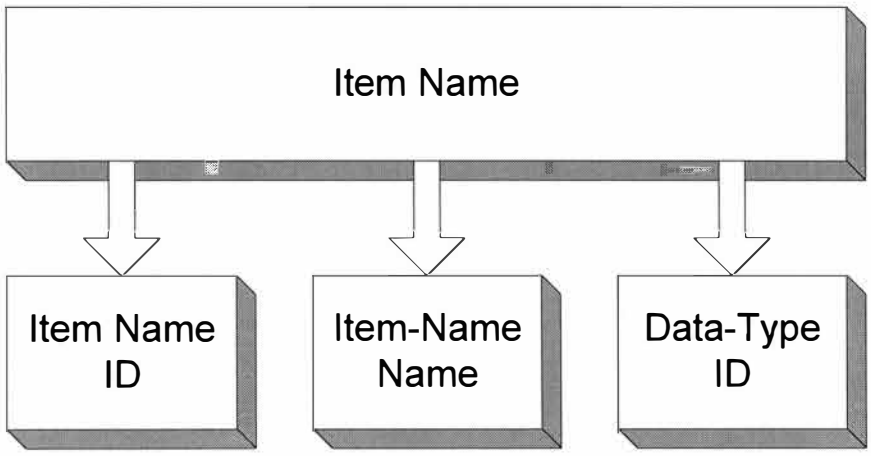

Figure 7. Item Name is Consist of Item Number, Item-Name Name, and Data Type ID.

Management Cost

For any property there is a management cost. The entity that collects rents,

contracts with vendors for services, prepares annual financial statement and generally oversees the asset management will be engaged. On small properties, this is done by the owner, but it is still and expense to be accounted for on each property. The management cost table is consists of management cost number, quantity, unit, unit cost, total cost, and name id. The management cost structure is shown in Figure 8.

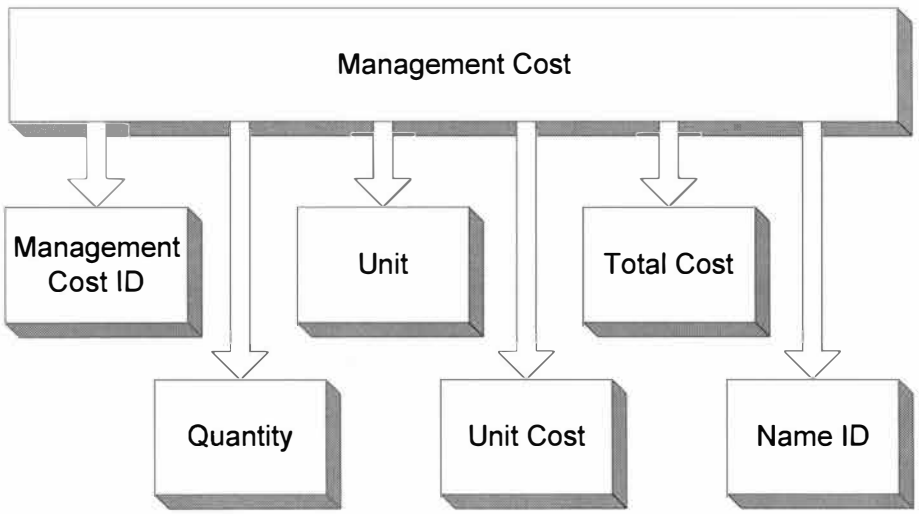

Figure 8. Management Cost Consist of Management Cost ID, Quantity, Unit, Unit Cost, Total Cost, and Name ID.

Operating Cost

Operating cost consists of utilities, maintenance, and repair costs. They are normally paid annually from the operating budget. The basic difference is that maintenance is usually scheduled in advance, whereas repairs are for items that break down or that require emergency action, such as broken pipes or failed motors. Both qualify for the general description of ordinary repairs. RSMeans (1996) The operating cost table consists of operating cost id, cost, unit, unit cost, date, and name id. The structure of the operating cost table is shown in Figure 9.

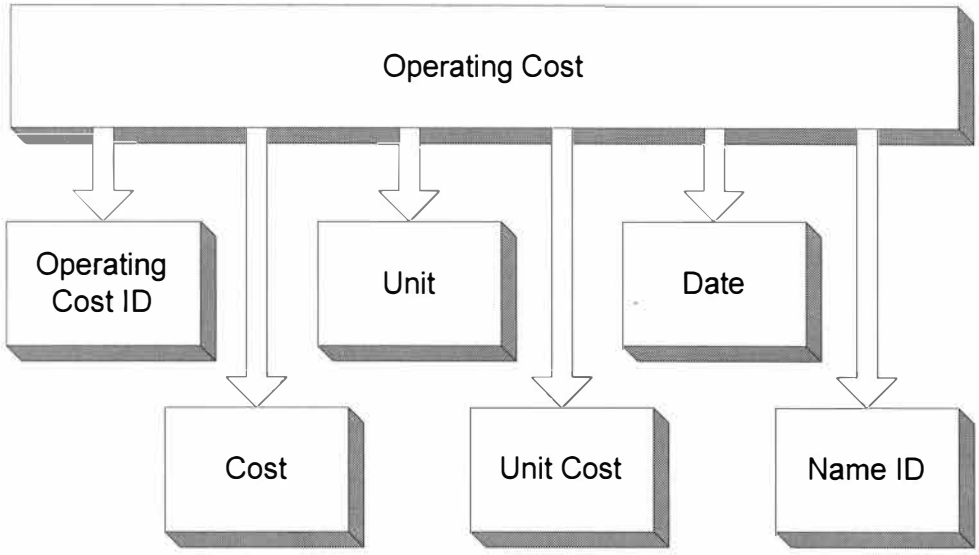

Figure 9. Operating Cost Consist of Operating Cost ID, Cost, Unit, Unit Cost, Date, and Name ID.

Maintenance Cost

Maintenance cost table as shown in Figure 10 consists of maintenance cost ID, description, cost, date, and name id. Description field is used for store description of maintenance in each item. Date field is used for record the maintenance date that the item has been maintenance.

The architecture of CFMMS is to connect each table together. The architecture of CFMMS is shown in Figure 11.

CFMMS Methodology

Designed Software

CFMMS software is developed by Microsoft Visual BASIC 6 that is in the Microsoft Visual Studio 6 Professional Version. Microsoft Visual Basic 6 can handle

the most complex database and can easily create software interface such in CFMMS. The CFMMS software has been developed by follow the step as shown in Figure 12.

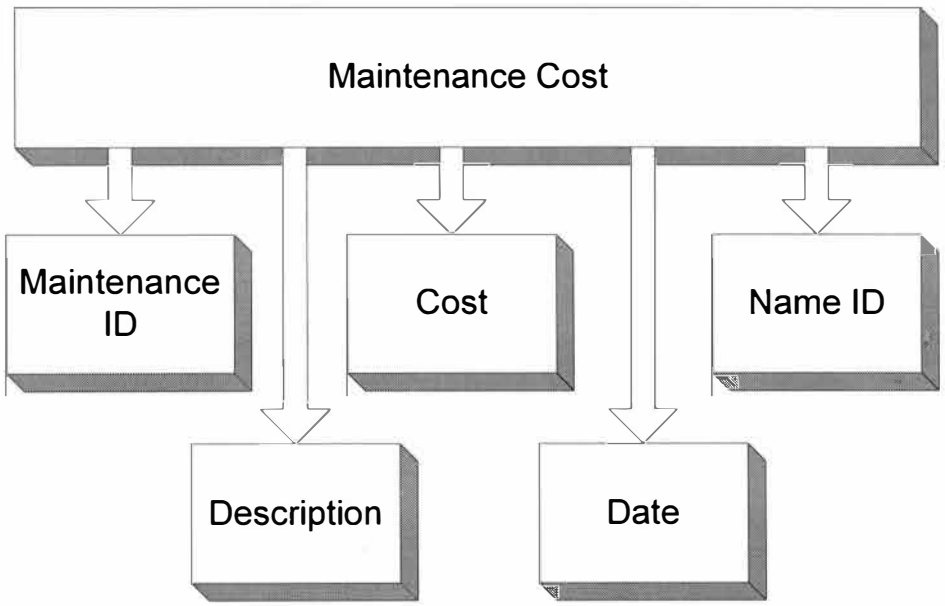

Figure 10. Maintenance Cost Consists of Maintenance ID, Descriptions of Maintenance, Cost, Data, and Name ID.

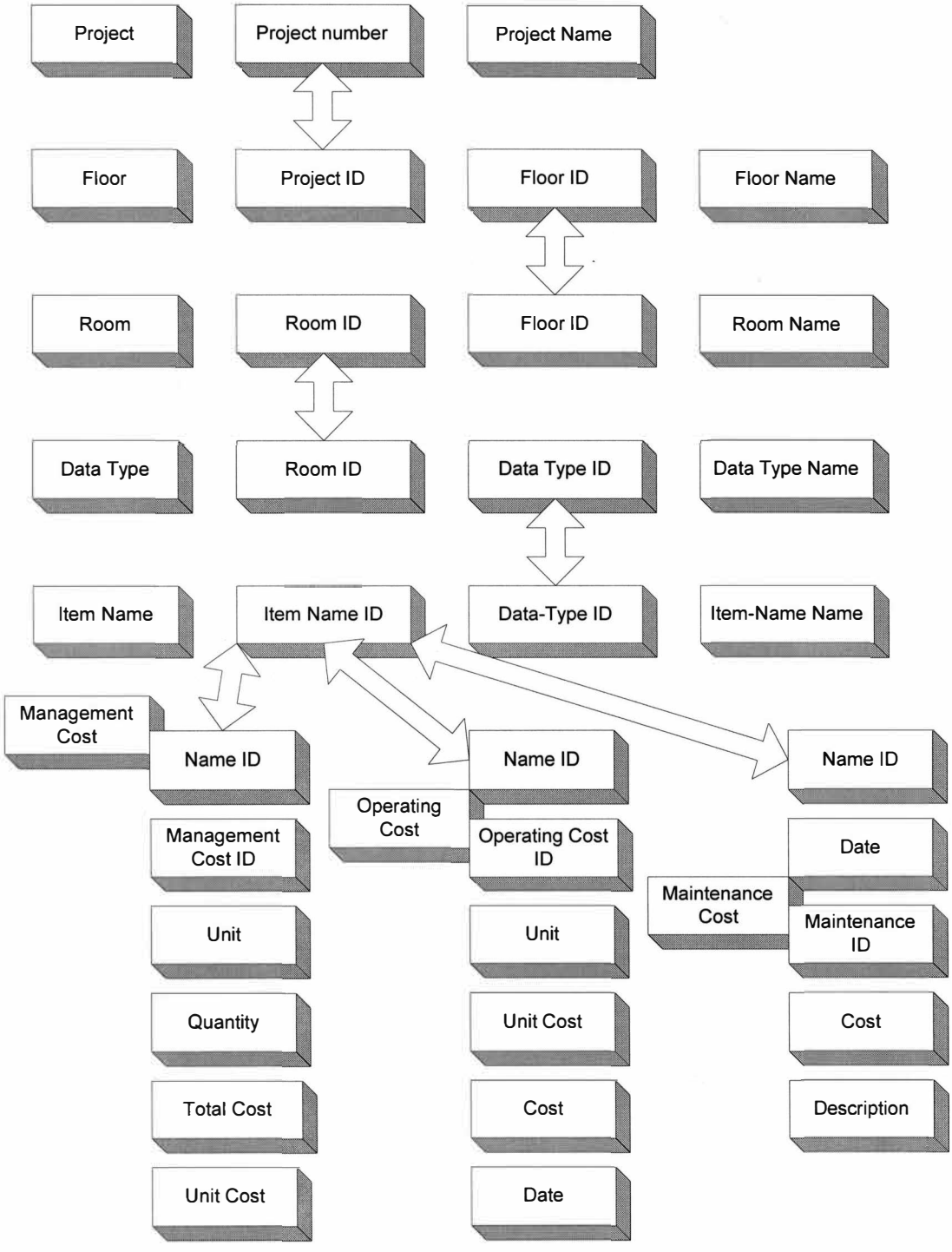

Figure 11. The Architecture of CFMMS.

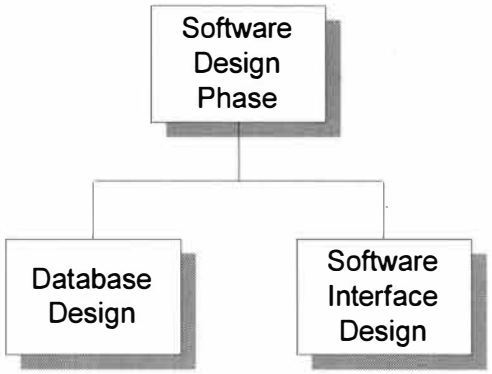

Figure 12. Software Design Phase.

Database Design

The database in software architecture is designed using Microsoft Access. The entire table is created and related to each other as shown in Figure 13.

As in Figure 13, every table is related to each other in the way of looking into the building. Project table and floor table are linked by project ID. Name table and operating cost table are linked by Name ID. This kind of relationship will make the user know exact place of the item.

Software Interface Design

The software interface design in Microsoft Visual Basic has several options to use. The method that has been chosen for CFMMS are pull-down menu and picture interface. The interface of CFMMS is shown in Figure 14.

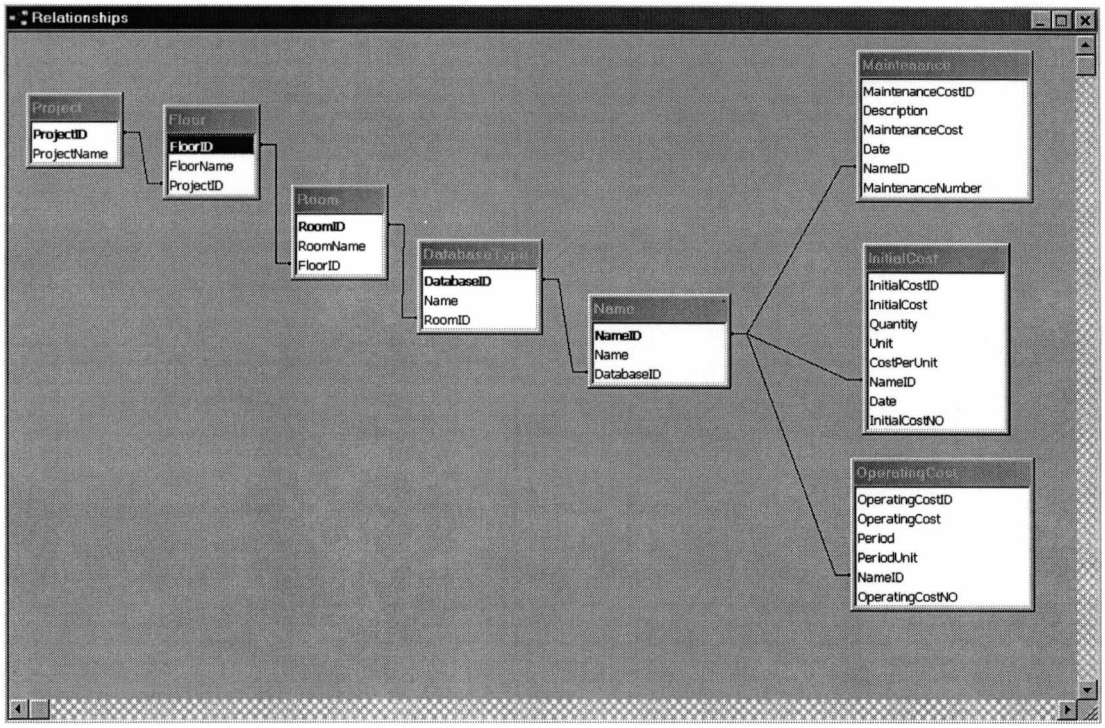

Figure 13. Data Relationships Using Microsoft Access Relationships Table.

Pull-down Menu is the interface that shows the entire list that show the existing database. This type of menu will be easily to choose data for data entry. The pull-down menu is shown in Figure 15.

Picture interface is a picture that can be use along with the pull-down menu to make it even more easily to get into the item that the user want. The example is shown in Figure 16.

Both interface code are shown in Appendix A. User have option to choose either interfaces depend on their preference.

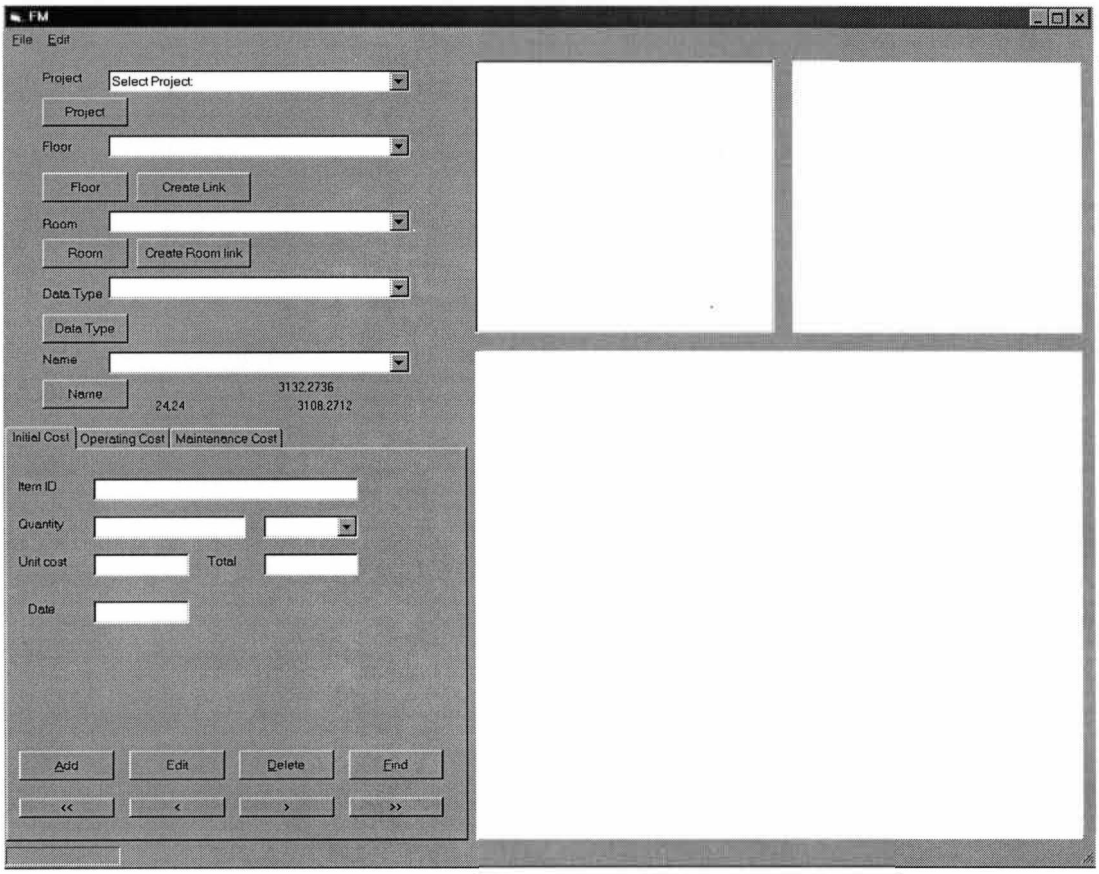

Figure 14. The CFMMS Program.

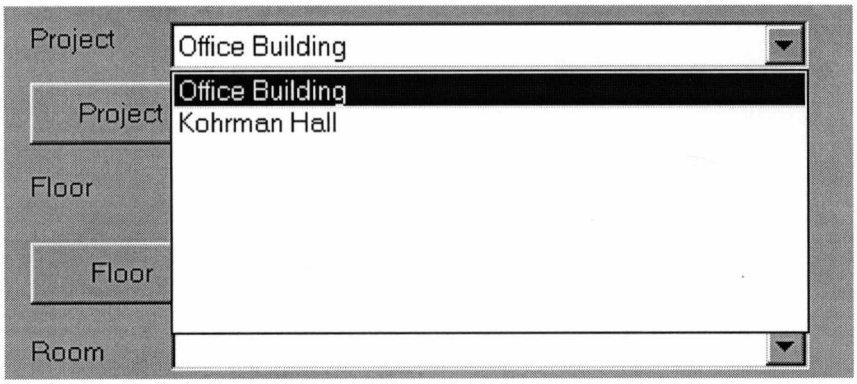

Figure 15. Pull Down Menu Example.

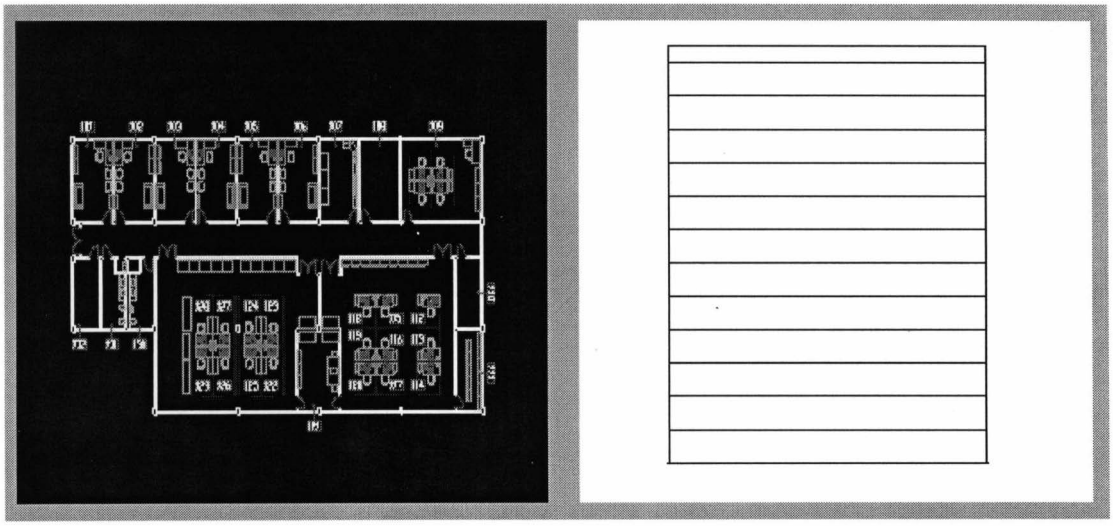

Figure 16. The Picture Interface Example.

CHAPTER VI

CFMMS PROGRAM

Computerized Facilities Maintenance Management Software

The CFMMS software interface is shown in Figure 17.

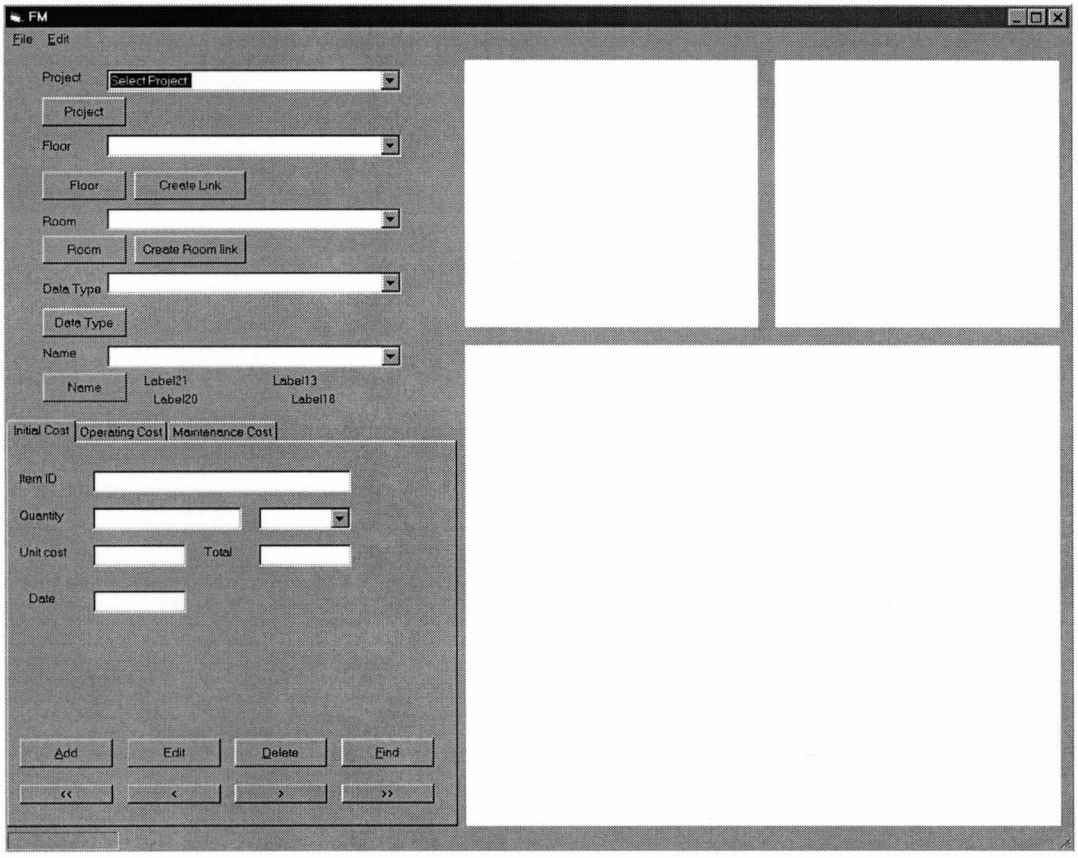

Figure 17. CFMMS Started Program.

Instruction on How to Use CFMMS

Creating New Project

1. Click on the project button. A dialog box of project management will show on screen as shown in Figure 18.

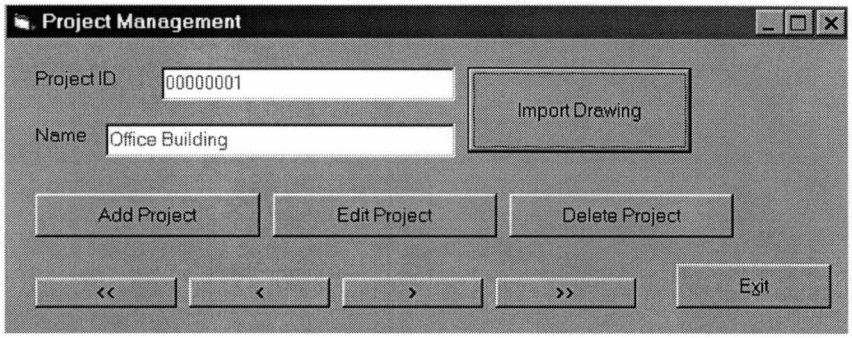

Figure 18. Project Management.

2. Click add project to add a new project as shown in Figure 19.

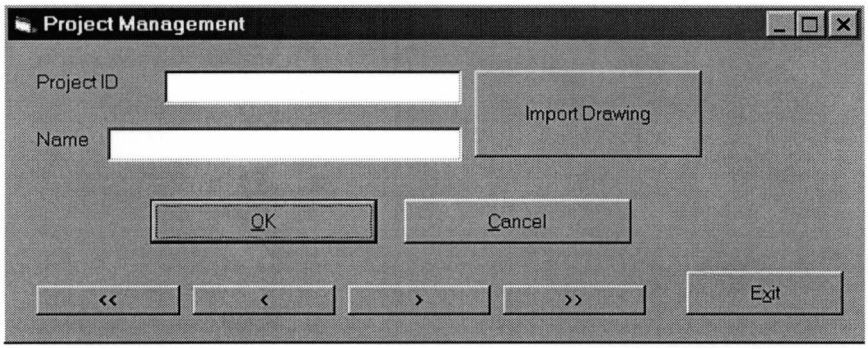

Figure 19. Adding Project Management Dialog Box.

3. Enter project ID and Name and then click OK. The new project has been added to the CFMMS.

4. Import drawing by clicking on the import drawing button. The open file dialog box will appear as shown in Figure 20.

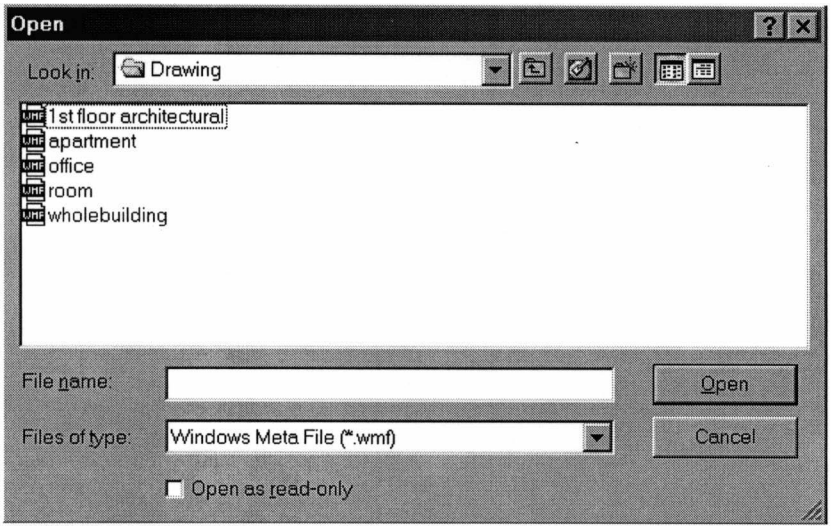

Figure 20. Import Drawing Dialog Box.

5. Choose the drawing file that corresponds to the project. Click OK.

6. Click Done to exit from project management.

Create New Floor

1. Click on the Floor button. The floor dialog box will be shown.

2. Follow the step 2 in create new project.

3. Click done.

Create Link of Floor to the Project

1. Click the Create Link button. The project-drawing picture will be selected.

2. Start drag from the top left of the floor drawing in the building picture to create the rectangular box that can be click to choose the floor and drop on the bottom

right corner of the floor.

3. The box that you create will appear on the picture that display the box that you just create.

4. Click OK if you want to use this box as a link area or Click NO to cancel the link.

Create New Room and Link Room to Floor

The method of creating new room and link is similar to create and link floor as shown in Figure 21.

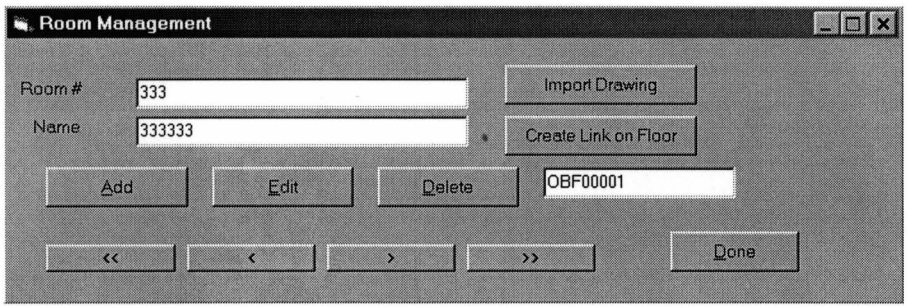

Figure 21. Room Management Dialog Box.

Create New Data Type

The data type interface is using pull down menu and picture menu linked from room drawing. The data type interface is shown in Figure 22.

Create New Name

The name interface is using only pull-down menu from room drawing. The name interface is shown in Figure 23.

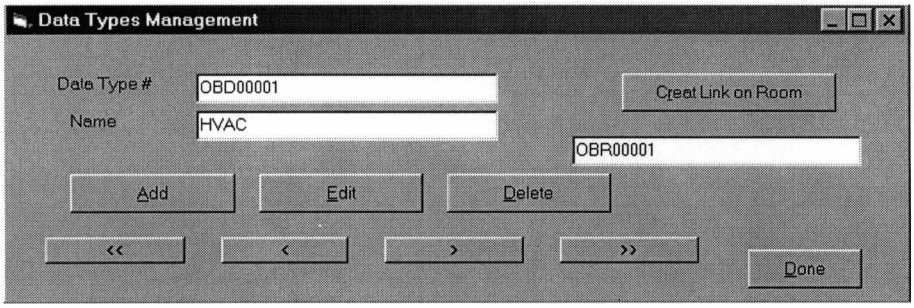

Figure 22. Data Types Management Dialog Box.

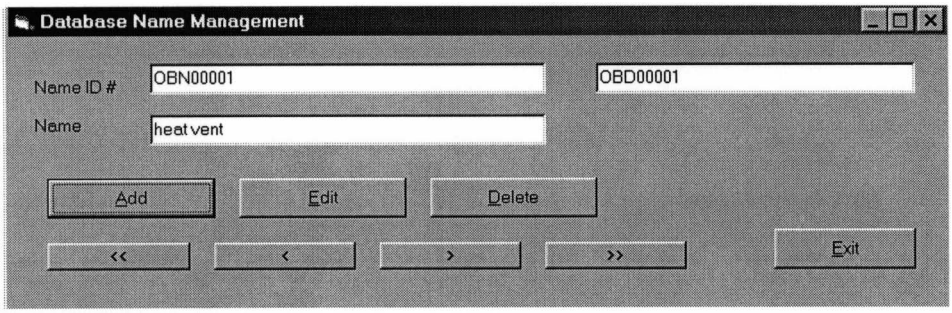

Figure 23. Database Name Management Dialog Box.

Management Cost Data

The management cost data is show in Figure 24. All buttons have the same function as the above database.

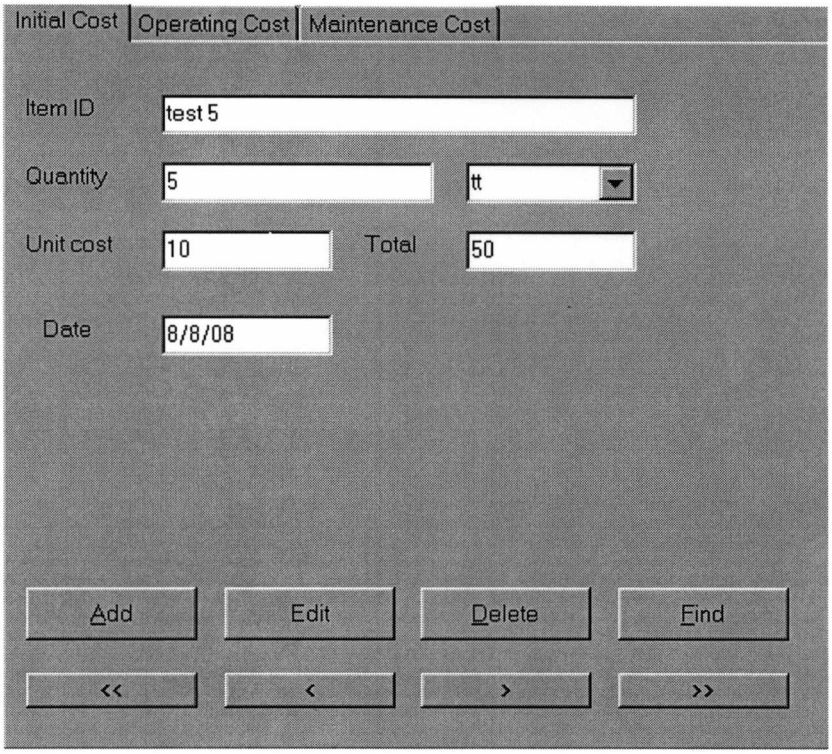

Figure 24. Initial Cost Database.

Operating Cost Data

The operating cost is the cost of operating the facilities. The operating cost is shown is Figure 25.

Maintenance Cost

The maintenance cost data is shown in Figure 26. To add new data click on add button. To edit data, click on the edit button. The four arrow buttons are used for select the data.

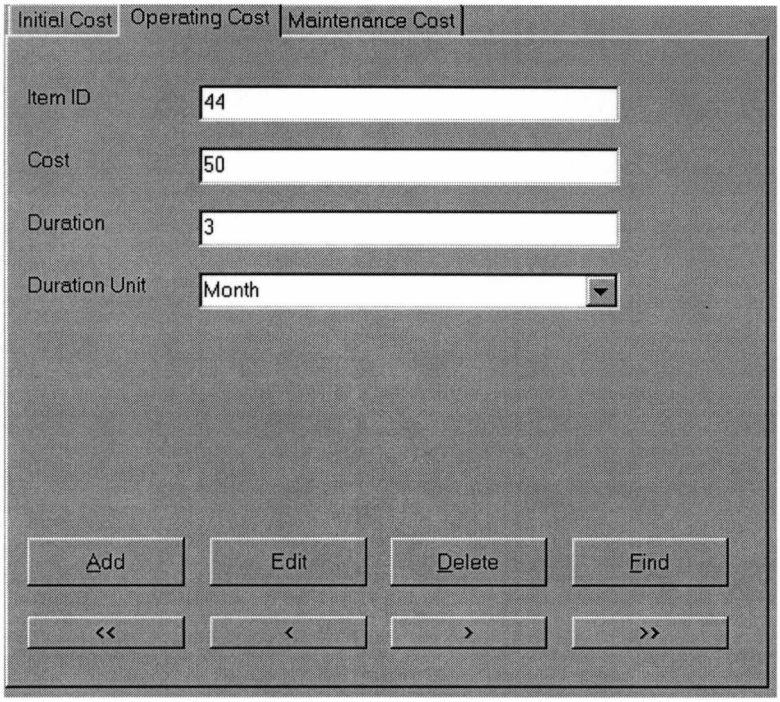

Figure 25. Operating Cost Database.

Summary

CFMMS program is a user-friendly program. User can use it even without manual just when they start to use it. The picture navigator is the fast way to get into the item that user want. The navigator of the cost data makes the program easy to access data.

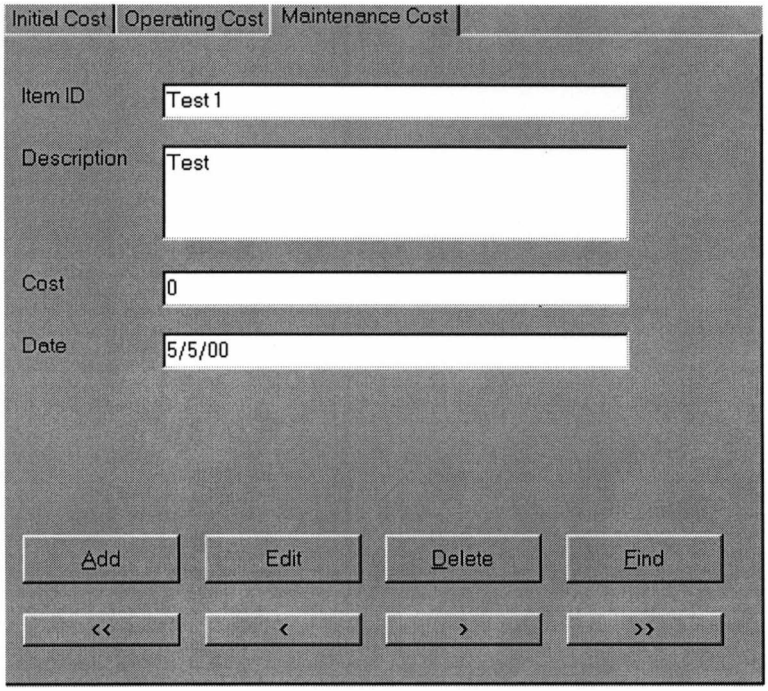

Figure 26. Maintenance Cost Database.

CHAPTER VII

CONCLUSION

Few people would dispute the importance of computers in business today. Computerization has led to increased efficiencies throughout the business world. Downsizing in the nineties has forced many facilities departments to do more with less, making tools for efficiency ever more essential. Many organizations have implemented computerization using a combination of spreadsheets, scheduling and word processing software to support the maintenance function. Today, Computerized Maintenance Management Systems are going far beyond these basic functions and integrating facility management into the company's MIS mainstream.

The Computerized Facilities Maintenance Management System (CFMMS) is developed by integrating windows application, Microsoft Visual Basic6, Access. The program has three important cost data which are (1) management cost data, (2) operating cost data, and (3) maintenance cost data. The system provides a point-and-click environment, so user does not have to find the application program to open the design data files. User interface make it easily to entry data that can be useful in the future cost data analysis.

CFMMS can be use in any facility that wants to replace the paper based cost data. The benefit of computer based over the paper based cost data are significantly save the time and money.

Appendix A

Partial Program Code Listing of CFMMS From Visual Basic 6 Program

MAIN PROGRAM

```
Private Sub Form_Load()
   DataEnvironment1.Projects
   Set RSProject = DataEnvironment1.rsProjects
   RSProject.Sort = "ProjectName"
   Set dcprojects.RowSource = RSProject
   dcprojects.ListField = "ProjectName"
   'Dim filenum As Integer
   'filenum = FreeFile
   'Open "c:\my thesis\picdat.dat" For Random As filenum
   'picDrawing.Picture = LoadPicture(apartment)
   'Tab
   Dim i As Integer
   For i = 0 To FrameData.count - 1
   With FrameData(i)
    Move TabData.ClientLeft, TabData.ClientTop, TabData.ClientWidth,
TabData.ClientHeight
      .Visible = False
      .BorderStyle = 0
   End With
   Next i
   CurrentTab = 1
   Set TabData.SelectedItem = TabData.Tabs(CurrentTab)
   FrameData(CurrentTab - 1).Visible = True
   InFindMode
   InMain
   'set mc, ic, opc rstemp
   If DataEnvironment1.rsEditIC.State = adStateOpenThen
DataEnvironment1.rsEditIC.Close
   On Error Resume Next
   DataEnvironment1.EditIC RSName.Fields("NameID")
   Set RSICTemp = DataEnvironment1.rsEditIC
   'Set data to be blank"
   txtItemNO.Text = ""
   txtQuantity.Text = ""
   txtUnitCost.Text = ""
   txtDate.Text = ""
   txtTotalcost.Text = ""
   cboUnit.Text = ""
   txtNameID = ""
   txtMItemNO.Text = ""
   txtMDescription.Text = ""
```

```
    txtMCost.Text = ""
    txtMDate.Text = ""
    txtOPItemNo.Text = ""
    txtOPCost.Text = ""
    txtOPDuration.Text = ""
    cboOPUnit.Text = ""
End Sub
Private Sub InvisibleButton1_Click()
End Sub
Private Sub Picture2_Click()
End Sub
Private Sub PictureNav_GotFocus()
    PictureNav.BorderStyle = 1
    FloorPicture.BorderStyle = 0
    RoomPicture.BorderStyle = 0
End Sub
    Private Sub PictureNav_MouseDown(Button As Integer, Shift As Integer, X As
Single, Y As Single)
    CurrentStartX = X
    CurrentStartY = Y
    Label13.Caption = X & "," & Y
End Sub
Private Sub PictureNav_MouseUp(Button As Integer, Shift As Integer, X As Single,
Y As Single)
    CurrentEndX = X
    CurrentEndY = Y
    'For Testing
    Label18.Caption = X & "," & Y
    CurrentWidth = Abs(CurrentStartX - CurrentEndX)
    CurrentHeight = Abs(CurrentStartY - CurrentEndY)
    Label20.Caption = CurrentWidth & "," & CurrentHeight
    'label21.Caption
    'Add Mode
    If AddMode And dcfloors.Text <> "Select Floor:" And dcfloors.Text <> "" Then
        'Draw a box of floor
        PictureNav.Line (CurrentStartX, CurrentStartY)-(CurrentEndX, CurrentEndY), ,
B
        Reply = MsgBox("Do you want to use the current area as a floor navigator?",
vbYesNo)
        If Reply = vbYes Then
            'add area data for link
            PositionName = dcfloors.Text
            PosProjectName = dcprojects.Text
            PosProjectPic = ProjectPic
```

```
            AddFloorBox
        End If
    'Find Mode
    Else
        FindFloorBox
        If CurrentFloor <> "" Then
            dcfloors.Text = CurrentFloor
        End If
        Label21.Caption = CurrentFloor
    End If
    InFindMode
    'delete the box
    PictureNav.Refresh
    PictureNav.AutoRedraw = False
End Sub
Private Sub RoomPicture_GotFocus()
    RoomPicture.BorderStyle = 1
    FloorPicture.BorderStyle = 0
    PictureNav.BorderStyle = 0
End Sub
Private Sub TabData_Click()

    If TabData.SelectedItem.Index = CurrentTab Then Exit Sub
    FrameData(TabData.SelectedItem.Index - 1).Visible = True
    FrameData(CurrentTab - 1).Visible = False
    CurrentTab = TabData.SelectedItem.Index

End Sub
Sub UnlockFields()
    'Initial Cost
    txtItemNO.Locked = False
    txtQuantity.Locked = False
    txtUnitCost.Locked = False
    txtTotalcost.Locked = False
    txtDate.Locked = False
End Sub
Sub UnlockFieldsM()
    'Maintenance Cost
    txtMItemNO.Locked = False
    txtMDescription.Locked = False
    txtMCost.Locked = False
    txtMDate.Locked = False
End Sub
Sub unlockfieldsO()
```

```
    'Operation Cost
    txtOPItemNo.Locked = False
    txtOPCost.Locked = False
    txtOPDuration.Locked = False
    cboOPUnit.Locked = False

End Sub
Sub LockFields()
    'Initial Cost
    txtItemNO.Locked = True
    txtQuantity.Locked = True
    txtUnitCost.Locked = True
    txtTotalcost.Locked = True
    txtDate.Locked = True
End Sub
Sub LockFieldsM()
    'Maintenance Cost
    txtMItemNO.Locked = True
    txtMDescription.Locked = True
    txtMCost.Locked = True
    txtMDate.Locked = True
End Sub
Sub LockFieldsO()
    'Operation Cost
    txtOPItemNo.Locked = True
    txtOPCost.Locked = True
    txtOPDuration.Locked = True
    cboOPUnit.Locked = True

End Sub
Sub HideButtons()
    'initial cost
    cmdOK.Visible = True
    cmdCancel.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
    cmdFind.Visible = False
End Sub
Sub HideButtonsM()
    'maintenance cost
    cmdOKMC.Visible = True
    cmdCancelMC.Visible = True
    cmdAddM.Visible = False
```

```
    cmdEditMC.Visible = False
    cmdDeleteMC.Visible = False
    cmdFindMC.Visible = False
End Sub
Sub HideButtonsO()
    'operating cost
    cmdOKOC.Visible = True
    cmdCancelOC.Visible = True
    cmdAddOC.Visible = False
    cmdEditOC.Visible = False
    cmdDeleteOC.Visible = False
    cmdFindOC.Visible = False
End Sub
Sub ShowButtons()
    'initialcost
    cmdOK.Visible = False
    cmdCancel.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
    cmdFind.Visible = True
End Sub
Sub ShowButtonsM()
    'maintenance cost
    cmdOKMC.Visible = False
    cmdCancelMC.Visible = False
    cmdAddM.Visible = True
    cmdEditMC.Visible = True
    cmdDeleteMC.Visible = True
    cmdFindMC.Visible = True
End Sub
Sub ShowButtonsO()
    'operatingcost
    cmdOKOC.Visible = False
    cmdCancelOC.Visible = False
    cmdAddOC.Visible = True
    cmdEditOC.Visible = True
    cmdDeleteOC.Visible = True
    cmdFindOC.Visible = True
End Sub
```

FLOOR

```
Private Sub Form_Load()
   Dim Filter As String
   Filter = "Windows Meta File (*.wmf)| *.wmf"
   Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
   Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
   Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
   Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
   Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
   Filter = Filter & "|Icons (*.ico)|*.ico"
   Filter = Filter & "|All files (*.*)| *.*"
   dlgimportFloor.Filter = Filter

   'InAddMode

End Sub
Sub UnlockFields()
   txtFloorName.Locked = False
   txtNumber.Locked = False
   txtProjectID.Locked = False
End Sub
Sub LockFields()
   txtFloorName.Locked = True
   txtNumber.Locked = True
   txtProjectID.Locked = True
End Sub
Sub HideButtons()
   cmdOKOK.Visible = True
   cmdCancell.Visible = True
   cmdAdd.Visible = False
   cmdEdit.Visible = False
   cmdDelete.Visible = False
End Sub
Sub ShowButtons()
   cmdOKOK.Visible = False
   cmdCancell.Visible = False
   cmdAdd.Visible = True
   cmdEdit.Visible = True
   cmdDelete.Visible = True
End Sub

Private Sub Form_Load()
```

```
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgimportFloor.Filter = Filter

    'InAddMode

End Sub
Sub UnlockFields()
    txtFloorName.Locked = False
    txtNumber.Locked = False
    txtProjectID.Locked = False
End Sub
Sub LockFields()
    txtFloorName.Locked = True
    txtNumber.Locked = True
    txtProjectID.Locked = True
End Sub
Sub HideButtons()
    cmdOKOK.Visible = True
    cmdCancell.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
End Sub
Sub ShowButtons()
    cmdOKOK.Visible = False
    cmdCancell.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
End Sub

Private Sub Form_Load()
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
```

```
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgimportFloor.Filter = Filter

    'InAddMode

End Sub
Sub UnlockFields()
    txtFloorName.Locked = False
    txtNumber.Locked = False
    txtProjectID.Locked = False
End Sub
Sub LockFields()
    txtFloorName.Locked = True
    txtNumber.Locked = True
    txtProjectID.Locked = True
End Sub
Sub HideButtons()
    cmdOKOK.Visible = True
    cmdCancell.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
End Sub
Sub ShowButtons()
    cmdOKOK.Visible = False
    cmdCancell.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
End Sub

Private Sub Form_Load()
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
```

```
    Filter = Filter & "|All files (*.*)| *.*"
    dlgimportFloor.Filter = Filter

    'InAddMode

End Sub
Sub UnlockFields()
    txtFloorName.Locked = False
    txtNumber.Locked = False
    txtProjectID.Locked = False
End Sub
Sub LockFields()
    txtFloorName.Locked = True
    txtNumber.Locked = True
    txtProjectID.Locked = True
End Sub
Sub HideButtons()
    cmdOKOK.Visible = True
    cmdCancell.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
End Sub
Sub ShowButtons()
    cmdOKOK.Visible = False
    cmdCancell.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
End Sub

Private Sub Form_Load()
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgimportFloor.Filter = Filter

    'InAddMode
```

```
End Sub
Sub UnlockFields()
   txtFloorName.Locked = False
   txtNumber.Locked = False
   txtProjectID.Locked = False
End Sub
Sub LockFields()
   txtFloorName.Locked = True
   txtNumber.Locked = True
   txtProjectID.Locked = True
End Sub
Sub HideButtons()
   cmdOKOK.Visible = True
   cmdCancell.Visible = True
   cmdAdd.Visible = False
   cmdEdit.Visible = False
   cmdDelete.Visible = False
End Sub
Sub ShowButtons()
   cmdOKOK.Visible = False
   cmdCancell.Visible = False
   cmdAdd.Visible = True
   cmdEdit.Visible = True
   cmdDelete.Visible = True
End Sub

Private Sub cmdCancell_Click()
   If DataEnvironment1.rsEditFloor.EditMode = adEditAdd Then
      DataEnvironment1.rsEditFloor.CancelUpdate
   Else
      DataEnvironment1.rsEditFloor.CancelUpdate
      DataEnvironment1.rsEditFloor.Move 0
   End If
   LockFields
   ShowButtons

   'Main.Show
   'NewProject.Hide
End Sub

Private Sub cmdDelete_Click()
   Reply = MsgBox("Record will be deleted permanently. Proceed?", vbYesNo)
   If Reply = vbYes Then
```

```
        On Error Resume Next
        DataEnvironment1.rsEditFloor.Delete
        If Err.Number <> 0 Then
           MsgBox "Could not delete record." & vbCrLf & Error.Description
           DataEnvironment1.rsEditFloor.CancelUpdate
        End If
        On Error GoTo 0
    End If
End Sub

Private Sub cmdEdit_Click()
    HideButtons
    UnlockFields
End Sub

Private Sub cmdFirst_Click()
    If RSFloorTemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditFloor.MoveFirst
        txtFloorName.Text = RSFloorTemp.Fields("floorname")
        txtNumber.Text = RSFloorTemp.Fields("floorid")
    End If
End Sub

Private Sub cmdImportFloor_Click()
    dlgimportFloor.ShowOpen
    'if user cancel
    If dlgimportFloor.FileName = "" Then Exit Sub
    Main.FloorPicture.Picture = LoadPicture(dlgimportFloor.FileName)

    PictureName = dlgimportFloor.FileName
    DatalinkName = txtFloorName.Text
    AddFloorDat
End Sub

Private Sub cmdLast_Click()
    If RSFloorTemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditFloor.MoveLast
        txtFloorName.Text = RSFloorTemp.Fields("floorname")
        txtNumber.Text = RSFloorTemp.Fields("floorid")
    End If
End Sub

Private Sub cmdNext_Click()
    If RSFloorTemp.RecordCount > 0 Then
```

```
      DataEnvironment1.rsEditFloor.MoveNext
        If DataEnvironment1.rsEditFloor.EOF Then
          DataEnvironment1.rsEditFloor.MoveLast
          'txtFloorName.Text = RSFloorTemp.Fields("floorname")
          'txtNumber.Text = RSFloorTemp.Fields("floorid")
        Else
          'txtFloorName.Text = RSFloorTemp.Fields("floorname")
          'txtNumber.Text = RSFloorTemp.Fields("floorid")
        End If
    txtFloorName.Text = RSFloorTemp.Fields("floorname")
    txtNumber.Text = RSFloorTemp.Fields("floorid")
    End If
End Sub

Private Sub cmdOK_Click()
    Main.Label21.Caption = CurrentFloor

    RSProject.Bookmark = Main.dcprojects.SelectedItem
    If DataEnvironment1.rsFloors.State = adStateOpen Then
DataEnvironment1.rsFloors.Close
    DataEnvironment1.floors RSProject.Fields("ProjectID")
    Set RSFloor = DataEnvironment1.rsFloors
    Set Main.dcfloors.RowSource = DataEnvironment1.rsFloors
    Main.dcfloors.ListField = "FloorName"
    Main.dcfloors.SetFocus
    Main.dcfloors.Text = "Select Floor:"

    AddFloorBox
    InMain
    InFindMode
End Sub

Private Sub cmdOKOK_Click()
    If txtFloorName.Text <> "" And txtNumber.Text <> "" Then
      'Add new floor to database
      On Error Resume Next
      DataEnvironment1.rsFloorMain.Update
      If Err.Number <> 0 Then
        If DataEnvironment1.rsEditFloor.EditMode = adEditAdd Then
          MsgBox "Could not add record. " & vbCrLf & Err.Description
        Else
          MsgBox "Could not save changes." & vbCrLf & Err.Description
        End If
        DataEnvironment1.rsEditFloor.CancelUpdate
```

```
            DataEnvironment1.rsEditFloor.Move 0
        End If
        ShowButtons
        LockFields
    Else
        MsgBox ("Please enter a floor name and floor ID")
    End If
End Sub

Private Sub cmdPickFloorArea_Click()
    Main.Show
    Main.PictureNav.SetFocus
    frmFloor.Hide
End Sub


Private Sub cmdPrevious_Click()
    If RSFloorTemp.RecordCount > 0 Then
        DataEnvironment1.rsEditFloor.MovePrevious
        If DataEnvironment1.rsEditFloor.BOF Then
            DataEnvironment1.rsEditFloor.MoveFirst
            txtFloorName.Text = RSFloorTemp.Fields("floorname")
            txtNumber.Text = RSFloorTemp.Fields("floorid")
        Else
            txtFloorName.Text = RSFloorTemp.Fields("floorname")
            txtNumber.Text = RSFloorTemp.Fields("floorid")
        End If

    End If
End Sub


Private Sub Form_Load()
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgimportFloor.Filter = Filter
```

```
    'InAddMode

End Sub
Sub UnlockFields()
    txtFloorName.Locked = False
    txtNumber.Locked = False
    txtProjectID.Locked = False
End Sub
Sub LockFields()
    txtFloorName.Locked = True
    txtNumber.Locked = True
    txtProjectID.Locked = True
End Sub
Sub HideButtons()
    cmdOKOK.Visible = True
    cmdCancell.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
End Sub
Sub ShowButtons()
    cmdOKOK.Visible = False
    cmdCancell.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
End Sub
```

ROOM

```
Private Sub Command3_Click()
   Me.Hide
   Main.Show
End Sub

Private Sub cmdADD_Click()
   HideButtons
   UnlockFields
   If RSRoomTemp.Supports(adBookmark) And RSRoomTemp.RecordCount > 0
Then
         lastRow = RSRoomTemp.Bookmark
   End If
   If DataEnvironment1.rsRoomMain.State = adStateClosed Then
DataEnvironment1.rsRoomMain.Open
   On Error Resume Next
   DataEnvironment1.rsRoomMain.AddNew
   txtFloorID.Text = RSFloor.Fields("FloorID")

End Sub

Private Sub cmdCancel1_Click()
   If DataEnvironment1.rsEditRoom.EditMode = adEditAdd Then
      DataEnvironment1.rsEditRoom.CancelUpdate
   Else
      DataEnvironment1.rsEditRoom.CancelUpdate
      DataEnvironment1.rsEditRoom.Move 0
   End If
   LockFields
   ShowButtons
End Sub

Private Sub cmdDelete_Click()
   Reply = MsgBox("Record will be deleted permanently. Proceed?", vbYesNo)
   If Reply = vbYes Then
      On Error Resume Next
      DataEnvironment1.rsEditRoom.Delete
      If Err.Number <> 0 Then
         MsgBox "Could not delete record." & vbCrLf & Error.Description
         DataEnvironment1.rsEditRoom.CancelUpdate
      End If
      On Error GoTo 0
```

```
    End If
End Sub

Private Sub cmdEdit_Click()
    HideButtons
    UnlockFields
End Sub

Private Sub cmdFirst_Click()
    If RSRoomTemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditRoom.MoveFirst
        txtName.Text = RSRoomTemp.Fields("RoomName")
        txtRoomID.Text = RSRoomTemp.Fields("RoomID")
    End If
End Sub

Private Sub cmdImport_Click()
    dlgImportRoom.ShowOpen
    'if user cancel
    If dlgImportRoom.FileName = "" Then Exit Sub
    Main.RoomPicture.Picture = LoadPicture(dlgImportRoom.FileName)

    PictureName = dlgImportRoom.FileName
    DatalinkName = txtName.Text
    AddFloorDat
End Sub

Private Sub cmdLast_Click()
    If RSRoomTemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditRoom.MoveLast
        txtName.Text = RSRoomTemp.Fields("RoomName")
        'error when it's on last record and click it
        txtRoomID.Text = RSFloorTemp.Fields("RoomID")
    End If
End Sub

Private Sub cmdNext_Click()
    If RSRoomTemp.RecordCount > 0 Then
        DataEnvironment1.rsEditRoom.MoveNext
            If DataEnvironment1.rsEditRoom.EOF Then
                DataEnvironment1.rsEditRoom.MoveLast
                'txtName.Text = RSroomTemp.Fields("RoomName")
                'txtRoomID.Text = RSroomTemp.Fields("RoomID")
            Else
```

```
            'txtName.Text = RSroomTemp.Fields("RoomName")
            'txtRoomID.Text = RSroomTemp.Fields("RoomID")
          End If
    txtName.Text = RSRoomTemp.Fields("RoomName")
    txtRoomID.Text = RSRoomTemp.Fields("RoomID")
    End If
End Sub

Private Sub cmdOK_Click()
    'Main.Label21.Caption = CurrentFloor

    RSFloor.Bookmark = Main.dcfloors.SelectedItem
    If DataEnvironment1.rsRooms.State = adStateOpen Then
DataEnvironment1.rsRooms.Close
    DataEnvironment1.rooms RSFloor.Fields("FloorID")
    Set RSRoom = DataEnvironment1.rsRooms
    Set Main.dcRooms.RowSource = DataEnvironment1.rsRooms
    Main.dcRooms.ListField = "RoomName"
    Main.dcRooms.SetFocus
    Main.dcRooms.Text = "Select Room:"

    AddFloorBox
    InMain
    InFindMode
End Sub

Private Sub cmdOKOK_Click()
    If txtName.Text <> "" And txtRoomID.Text <> "" Then
        'Add new room to database
        On Error Resume Next
        DataEnvironment1.rsRoomMain.Update
        If Err.Number <> 0 Then
            If DataEnvironment1.rsEditRoom.EditMode = adEditAdd Then
                MsgBox "Could not add record. " & vbCrLf & Err.Description
            Else
                MsgBox "Could not save changes." & vbCrLf & Err.Description
            End If
            DataEnvironment1.rsEditRoom.CancelUpdate
            DataEnvironment1.rsEditRoom.Move 0
        End If
        ShowButtons
        LockFields
    Else
        MsgBox ("Please enter a room name and room ID")
```

```
    End If
End Sub

Private Sub cmdPickRoom_Click()
    'need to add data to link
End Sub

Private Sub cmdPrevious_Click()
    If RSRoomTemp.RecordCount > 0 Then
        DataEnvironment1.rsEditRoom.MovePrevious
        If DataEnvironment1.rsEditRoom.BOF Then
            DataEnvironment1.rsEditRoom.MoveFirst
            txtName.Text = RSRoomTemp.Fields("RoomName")
            txtRoomID.Text = RSRoomTemp.Fields("RoomID")
        Else
            txtName.Text = RSRoomTemp.Fields("RoomName")
            txtRoomID.Text = RSRoomTemp.Fields("RoomID")
        End If

    End If
End Sub

Private Sub Form_Load()
    Dim Filter As String
    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgImportRoom.Filter = Filter

    InAddMode
End Sub
Sub UnlockFields()
    txtName.Locked = False
    txtRoomID.Locked = False
    txtFloorID.Locked = False
End Sub
Sub LockFields()
    txtName.Locked = True
    txtRoomID.Locked = True
```

```
    txtFloorID.Locked = True
End Sub
Sub HideButtons()
    cmdOKOK.Visible = True
    cmdCancell.Visible = True
    cmdAdd.Visible = False
    cmdEdit.Visible = False
    cmdDelete.Visible = False
End Sub
Sub ShowButtons()
    cmdOKOK.Visible = False
    cmdCancell.Visible = False
    cmdAdd.Visible = True
    cmdEdit.Visible = True
    cmdDelete.Visible = True
End Sub
```

DATA TYPE

```
Private Sub cmdADD_Click()
   HideButtons
   UnlockFields
   If RSDATATemp.Supports(adBookmark) And RSDATATemp.RecordCount > 0
Then
         lastRow = RSDATATemp.Bookmark
   End If
   If DataEnvironment1.rsDataMain.State = adStateClosed Then
DataEnvironment1.rsDataMain.Open
   On Error Resume Next
   DataEnvironment1.rsDataMain.AddNew
   txtRoomID.Text = RSRoom.Fields("RoomID")
End Sub
Sub UnlockFields()
   txtName.Locked = False
   txtID.Locked = False
   txtRoomID.Locked = False
End Sub
Sub LockFields()
   txtName.Locked = True
   txtID.Locked = True
   txtRoomID.Locked = True
End Sub
Sub HideButtons()
   cmdOKOK.Visible = True
   cmdCancel.Visible = True
   cmdAdd.Visible = False
   cmdEdit.Visible = False
   cmdDelete.Visible = False
End Sub
Sub ShowButtons()
   cmdOKOK.Visible = False
   cmdCancel.Visible = False
   cmdAdd.Visible = True
   cmdEdit.Visible = True
   cmdDelete.Visible = True
End Sub

Private Sub cmdCancel_Click()
   If DataEnvironment1.rsEditData.EditMode = adEditAdd Then
      DataEnvironment1.rsEditData.CancelUpdate
```

```
    Else
        DataEnvironment1.rsEditData.CancelUpdate
        DataEnvironment1.rsEditData.Move 0
    End If
    LockFields
    ShowButtons
End Sub

Private Sub cmdDelete_Click()
    Reply = MsgBox("Record will be deleted permanently. Proceed?", vbYesNo)
    If Reply = vbYes Then
        On Error Resume Next
        DataEnvironment1.rsEditData.Delete
        If Err.Number <> 0 Then
            MsgBox "Could not delete record." & vbCrLf & Error.Description
            DataEnvironment1.rsEditData.CancelUpdate
        End If
        On Error GoTo 0
    End If
End Sub

Private Sub cmdEdit_Click()
    HideButtons
    UnlockFields
End Sub

Private Sub cmdFirst_Click()
    If RSDATATemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditData.MoveFirst
        txtName.Text = RSDATATemp.Fields("Name")
        txtID.Text = RSDATATemp.Fields("DatabaseID")
    End If
End Sub

Private Sub cmdLast_Click()
    If RSDATATemp.RecordCount <> 0 Then
        DataEnvironment1.rsEditData.MoveLast
        txtName.Text = RSDATATemp.Fields("Name")
        txtID.Text = RSDATATemp.Fields("databaseID")
    End If

End Sub

Private Sub cmdNext_Click()
```

```
    If RSDATATemp.RecordCount > 0 Then
      DataEnvironment1.rsEditData.MoveNext
        If DataEnvironment1.rsEditData.EOF Then
          DataEnvironment1.rsEditData.MoveLast
          'txtName.Text = RSroomTemp.Fields("RoomName")
          'txtRoomID.Text = RSroomTemp.Fields("RoomID")
        Else
          'txtName.Text = RSroomTemp.Fields("RoomName")
          'txtRoomID.Text = RSroomTemp.Fields("RoomID")
        End If
    txtName.Text = RSDATATemp.Fields("Name")
    txtID.Text = RSDATATemp.Fields("DatabaseID")
    End If
End Sub

Private Sub cmdOK_Click()
    'Main.Label21.Caption = CurrentFloor

    RSRoom.Bookmark = Main.dcRooms.SelectedItem
    If DataEnvironment1.rsDatabasetypes.State = adStateOpen Then
DataEnvironment1.rsDatabasetypes.Close
    DataEnvironment1.databasetypes RSRoom.Fields("RoomID")
    Set RSdata = DataEnvironment1.rsDatabasetypes
    Set Main.dcDatabasetypes.RowSource = DataEnvironment1.rsDatabasetypes
    Main.dcDatabasetypes.ListField = "Name"
    Main.dcDatabasetypes.SetFocus
    Main.dcDatabasetypes.Text = "Select Databasetype:"

    AddFloorBox
    InMain
    InFindMode
End Sub

Private Sub cmdOKOK_Click()
    If txtName.Text <> "" And txtID.Text <> "" Then
      'Add new room to database
      On Error Resume Next
      DataEnvironment1.rsDataMain.Update
      If Err.Number <> 0 Then
        If DataEnvironment1.rsEditData.EditMode = adEditAdd Then
          MsgBox "Could not add record. " & vbCrLf & Err.Description
        Else
          MsgBox "Could not save changes." & vbCrLf & Err.Description
        End If
```

```
        DataEnvironment1.rsEditData.CancelUpdate
        DataEnvironment1.rsEditData.Move 0
      End If
      ShowButtons
      LockFields
    Else
      MsgBox ("Please enter a data type and data ID")
    End If
End Sub

Private Sub cmdPrevious_Click()
    If RSDATATemp.RecordCount > 0 Then
      DataEnvironment1.rsEditData.MovePrevious
      If DataEnvironment1.rsEditData.BOF Then
        DataEnvironment1.rsEditData.MoveFirst
        txtName.Text = RSDATATemp.Fields("Name")
        txtID.Text = RSDATATemp.Fields("DatabaseID")
      Else
        txtName.Text = RSDATATemp.Fields("Name")
        txtID.Text = RSDATATemp.Fields("databaseID")
      End If

    End If
End Sub
```

ITEM NAME

```
Private Sub cdmLast_Click()
   If RSNameTemp.RecordCount <> 0 Then
      DataEnvironment1.rsEditName.MoveLast
      txtName.Text = RSNameTemp.Fields("Name")
      txtID.Text = RSNameTemp.Fields("NameID")
   End If
End Sub

Private Sub cdmNext_Click()
   If RSNameTemp.RecordCount > 0 Then
      DataEnvironment1.rsEditName.MoveNext
         If DataEnvironment1.rsEditName.EOF Then
            DataEnvironment1.rsEditName.MoveLast
            'txtName.Text = RSroomTemp.Fields("RoomName")
            'txtRoomID.Text = RSroomTemp.Fields("RoomID")
         Else
            'txtName.Text = RSroomTemp.Fields("RoomName")
            'txtRoomID.Text = RSroomTemp.Fields("RoomID")
         End If
   txtName.Text = RSNameTemp.Fields("Name")
   txtID.Text = RSNameTemp.Fields("NameID")
   End If
End Sub

Private Sub cmdADD_Click()
   HideButtons
   UnlockFields
   If RSNameTemp.Supports(adBookmark) And RSNameTemp.RecordCount > 0
Then
      lastRow = RSNameTemp.Bookmark
   End If
   If DataEnvironment1.rsNameMain.State = adStateClosed Then
DataEnvironment1.rsNameMain.Open
   On Error Resume Next
   DataEnvironment1.rsNameMain.AddNew
   txtDataID.Text = RSdata.Fields("DatabaseID")
End Sub

Private Sub cmdCancel_Click()
   If DataEnvironment1.rsEditName.EditMode = adEditAdd Then
      DataEnvironment1.rsEditName.CancelUpdate
```

```
    Else
        DataEnvironment1.rsEditName.CancelUpdate
        On Error Resume Next
        DataEnvironment1.rsEditName.Move 0
    End If
    LockFields
    ShowButtons
End Sub

Private Sub cmdDelete_Click()
    Reply = MsgBox("Record will be deleted permanently. Proceed?", vbYesNo)
    If Reply = vbYes Then
        On Error Resume Next
        DataEnvironment1.rsEditName.Delete
        If Err.Number <> 0 Then
            MsgBox "Could not delete record." & vbCrLf & Error.Description
            DataEnvironment1.rsEditName.CancelUpdate
        End If
        On Error GoTo 0
    End If
End Sub

Private Sub cmdDone_Click()
    'Main.Label21.Caption = CurrentFloor

    'RSdata.Bookmark = Main.dcDatabasetypes.SelectedItem
    If DataEnvironment1.rsAllNames.State = adStateOpen Then
DataEnvironment1.rsAllNames.Close
    DataEnvironment1.allnames RSdata.Fields("DatabaseID")
    Set RSName = DataEnvironment1.rsAllNames
    Set Main.dcAllName.RowSource = DataEnvironment1.rsAllNames
    Main.dcAllName.ListField = "Name"
    Main.dcAllName.SetFocus
    Main.dcAllName.Text = "Select Item:"

    AddFloorBox
    InMain
    InFindMode
End Sub

Private Sub cmdEdit_Click()
    HideButtons
    UnlockFields
End Sub
```

```
Private Sub cmdFirst_Click()
   If RSNameTemp.RecordCount <> 0 Then
      DataEnvironment1.rsEditName.MoveFirst
      txtName.Text = RSNameTemp.Fields("Name")
      txtID.Text = RSNameTemp.Fields("NameID")
   End If
End Sub

Private Sub cmdOK_Click()
   If txtName.Text <> "" And txtID.Text <> "" Then
      'Add new room to database
      On Error Resume Next
      DataEnvironment1.rsNameMain.Update
      If Err.Number <> 0 Then
         If DataEnvironment1.rsEditName.EditMode = adEditAdd Then
            MsgBox "Could not add record. " & vbCrLf & Err.Description
         Else
            MsgBox "Could not save changes." & vbCrLf & Err.Description
         End If
         DataEnvironment1.rsEditName.CancelUpdate
         DataEnvironment1.rsEditName.Move 0
      End If
      ShowButtons
      LockFields
   Else
      MsgBox ("Please enter a name or name ID")
   End If
End Sub

Private Sub cmdPrevious_Click()
    If RSNameTemp.RecordCount > 0 Then
      DataEnvironment1.rsEditName.MovePrevious
      If DataEnvironment1.rsEditName.BOF Then
         DataEnvironment1.rsEditName.MoveFirst
         txtName.Text = RSNameTemp.Fields("Name")
         txtID.Text = RSNameTemp.Fields("NameID")
      Else
         txtName.Text = RSNameTemp.Fields("Name")
         txtID.Text = RSNameTemp.Fields("NameID")
      End If

   End If
End Sub
```

```
Sub UnlockFields()
   txtName.Locked = False
   txtID.Locked = False
   txtDataID.Locked = False
End Sub
Sub LockFields()
   txtName.Locked = True
   txtID.Locked = True
   txtDataID.Locked = True
End Sub
Sub HideButtons()
   cmdOK.Visible = True
   cmdCancel.Visible = True
   cmdADD.Visible = False
   cmdEdit.Visible = False
   cmdDelete.Visible = False
End Sub
Sub ShowButtons()
   cmdOK.Visible = False
   cmdCancel.Visible = False
   cmdADD.Visible = True
   cmdEdit.Visible = True
   cmdDelete.Visible = True
End Sub
```

PROJECT

```
Dim lastRow
Private Sub cmdAddProject_Click()
   HideButtons
   UnlockFields
   If DataEnvironment1.rsProjects.Supports(adBookmark) Then
         lastRow = DataEnvironment1.rsProjects.Bookmark
   End If
   DataEnvironment1.rsProjects.AddNew
End Sub

Private Sub cmdCancelNewProject_Click()
   If DataEnvironment1.rsProjects.EditMode = adEditAdd Then
      DataEnvironment1.rsProjects.CancelUpdate
   Else
      DataEnvironment1.rsProjects.CancelUpdate
      DataEnvironment1.rsProjects.Move 0
   End If
   LockFields
   ShowButtons

   'Main.Show
   'NewProject.Hide
End Sub

Private Sub cmddeleteproject_Click()
   Reply = MsgBox("Record will be deleted permanently. Proceed?", vbYesNo)
   If Reply = vbYes Then
      On Error Resume Next
      DataEnvironment1.rsProjects.Delete
      If Err.Number <> 0 Then
         MsgBox "Could not delete record." & vbCrLf & Error.Description
         DataEnvironment1.rsProjects.CancelUpdate
      End If
      On Error GoTo 0
   End If
End Sub

Private Sub cmdEditProject_Click()
   HideButtons
   UnlockFields
End Sub
```

```
Private Sub cmdExit_Click()
   Main.Show
   NewProject.Hide
End Sub

Private Sub cmdFirst_Click()
   DataEnvironment1.rsProjects.MoveFirst
End Sub

Private Sub cmdInsertProjectDwg_Click()
   dlgInsertDrawing.ShowOpen
   'if user cancel
   If dlgInsertDrawing.FileName = "" Then Exit Sub

   Main.PictureNav.Picture = LoadPicture(dlgInsertDrawing.FileName)

   PictureName = dlgInsertDrawing.FileName
   DatalinkName = txtNewProjectName.Text
   AddPicDat

End Sub

Private Sub cmdLast_Click()
   DataEnvironment1.rsProjects.MoveLast
End Sub

Private Sub cmdNext_Click()
   DataEnvironment1.rsProjects.MoveNext
   If DataEnvironment1.rsProjects.EOF Then
      DataEnvironment1.rsProjects.MoveLast
   End If
End Sub

Private Sub cmdOKNewProject_Click()
   If txtNewProjectName.Text <> "" Then
      'Add new project to database
      On Error Resume Next
      DataEnvironment1.rsProjects.Update
      If Err.Number <> 0 Then
         If DataEnvironment1.rsProjects.EditMode = adEditAdd Then
            MsgBox "Could not add record. " & vbCrLf & Err.Description
         Else
            MsgBox "Could not save changes." & vbCrLf & Err.Description
```

```
            End If
            DataEnvironment1.rsProjects.CancelUpdate
            DataEnvironment1.rsProjects.Move 0
        End If
        ShowButtons
        LockFields


        Main.Show
        NewProject.Hide
        Main.PictureNav.SetFocus
    Else
        MsgBox ("Please enter a project name")
    End If

End Sub

Private Sub cmdPrevious_Click()
    DataEnvironment1.rsProjects.MovePrevious
    If DataEnvironment1.rsProjects.BOF Then
        DataEnvironment1.rsProjects.MoveFirst
    End If
End Sub

Private Sub Form_Load()
    Dim Filter As String

    Filter = "Windows Meta File (*.wmf)| *.wmf"
    Filter = Filter & "|Bitmap (*.bmp)| *.bmp"
    Filter = Filter & "|Graphics Interachange Format (*.gif) | *.gif"
    Filter = Filter & "|Joint Photographic Experts Group (*.jpg) | *.jpg"
    Filter = Filter & "|Device Independent Bitmap (*.dib) | *.dib"
    Filter = Filter & "|Enhanced Meta File (*.emf) | *.emf"
    Filter = Filter & "|Icons (*.ico)|*.ico"
    Filter = Filter & "|All files (*.*)| *.*"
    dlgInsertDrawing.Filter = Filter

    LockFields

End Sub
Sub UnlockFields()
    txtNewProjectName.Locked = False
    txtProjectID.Locked = False
    txtNewProjectName.Enabled = True
```

```
    txtProjectID.Enabled = True
End Sub
Sub LockFields()
    txtNewProjectName.Locked = True
    txtProjectID.Locked = True
    txtNewProjectName.Enabled = False
    txtProjectID.Enabled = False
End Sub
Sub HideButtons()
    cmdOKNewProject.Visible = True
    cmdCancelNewProject.Visible = True
    cmdAddProject.Visible = False
    cmdEditProject.Visible = False
    cmddeleteproject.Visible = False
End Sub
Sub ShowButtons()
    cmdOKNewProject.Visible = False
    cmdCancelNewProject.Visible = False
    cmdAddProject.Visible = True
    cmdEditProject.Visible = True
    cmddeleteproject.Visible = True
End Sub
```

DATA MODULE

```
Public ProjectIDBM As String
Public FloorBox(1000, 1000) As String
Public RoomBox(1000, 1000) As String
Public DataTypeBox(1000, 1000) As String

Public RSProject As New ADODB.Recordset, RSName As New ADODB.Recordset
Public RSFloor As New ADODB.Recordset, RSRoom As New ADODB.Recordset,
RSdata As New ADODB.Recordset
Public RSMaintenanceCost As New ADODB.Recordset, RSInitialCost As New
ADODB.Recordset, RSOperatingCost As New ADODB.Recordset

Public FloorShowed As Boolean, RoomShowed As Boolean, NameShowed As
Boolean
Public MainShowed As Boolean, DataTypeShowed As Boolean
Public ProjectShowed As Boolean

Public RSFloorTemp As New ADODB.Recordset
Public RSRoomTemp As New ADODB.Recordset
Public RSDATATemp As New ADODB.Recordset
Public RSNameTemp As New ADODB.Recordset
Public RSMCTemp As New ADODB.Recordset
Public RSOPCTemp As New ADODB.Recordset
Public RSICTemp As New ADODB.Recordset

Public AddMode As Boolean
Public FindMode As Boolean
Public CurrentFloor As String
'Picture
Public CurrentStartX As Double, CurrentStartY As Double, CurrentEndX As Double,
CurrentEndY As Double
Public CurrentWidth As Double, CurrentHeight As Double

'Picture to show
Type Datfile
   DatName As String
   PicName As String
End Type
Public PictureData(1000) As Datfile
Public CurPicDat As Integer
Public DatalinkName As String, PictureName As String
Public DataRepeat As Integer
```

```
Public ProjectPic As String
Public FloorPic As String
Public Roompic As String

'Position of link
Type Posfile
   Posname As String
   PosSX As Integer
   PosEX As Integer
   PosSY As Integer
   PosEY As Integer
   PosPjName As String
   PosPjPic As String
   PosFlName As String
   PosFlPic As String
   PosRmName As String
   posRmPic As String
End Type
Public PositionData(1000) As Posfile
Public PositionName As String
Public PositionRepeat As Integer
Public PosProjectName As String
Public PosProjectPic As String


'Private variable
Private SX As Integer
Private SY As Integer
Private EX As Integer
Private EY As Integer
Private CX As Integer
Private CY As Integer
Sub AddPicDat()
   Dim filenum As Integer
   Dim info(1000) As String
   Dim count As Integer

   filenum = FreeFile
   Open "c:\my thesis\picdat.dat" For Input As filenum
   count = 0
   'record all data
   While Not EOF(filenum)
      Input #filenum, PictureData(count).DatName, PictureData(count).PicName
      count = count + 1
```

```
    Wend
    'verify existing dataname
    For i = 0 To count
       If PictureData(i).DatName = DatalinkName Or i = count Then
          DataRepeat = i
          i = count
       End If
    Next i
    PictureData(DataRepeat).DatName = DatalinkName
    PictureData(DataRepeat).PicName = PictureName

    Close #filenum

    If DataRepeat <> count Then count = count - 1
    'write data
    Open "c:\my thesis\picdat.dat" For Output As filenum
    For i = 0 To count
       Write #filenum, PictureData(i).DatName, PictureData(i).PicName

    Next i
    Close #filenum
End Sub
Sub AddFloorDat()
    Dim filenum As Integer
    Dim info(1000) As String
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\picdat.dat" For Input As filenum
    count = 0
    'record all data
    While Not EOF(filenum)
       Input #filenum, PictureData(count).DatName, PictureData(count).PicName
       count = count + 1
    Wend

    'verify existing dataname

    For i = 0 To count
       If PictureData(i).DatName = DatalinkName Or i = count Then
          DataRepeat = i
          i = count
       End If
    Next i
```

```
    PictureData(DataRepeat).DatName = DatalinkName
    PictureData(DataRepeat).PicName = PictureName

    Close #filenum

    If DataRepeat <> count Then count = count - 1
    'write data
    Open "c:\my thesis\picdat.dat" For Output As filenum
    For i = 0 To count
        Write #filenum, PictureData(i).DatName, PictureData(i).PicName

    Next i
    Close #filenum
End Sub
Sub AddRoomDat()
    Dim filenum As Integer
    Dim info(1000) As String
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\picdat.dat" For Input As filenum
    count = 0
    'record all data from file
    While Not EOF(filenum)
        Input #filenum, PictureData(count).DatName, PictureData(count).PicName
        count = count + 1
    Wend

    'verify existing dataname

    For i = 0 To count
        If PictureData(i).DatName = DatalinkName Or i = count Then
            DataRepeat = i
            i = count
        End If
    Next i
    PictureData(DataRepeat).DatName = DatalinkName
    PictureData(DataRepeat).PicName = PictureName

    Close #filenum

    If DataRepeat <> count Then count = count - 1
    'write data
    Open "c:\my thesis\picdat.dat" For Output As filenum
```

```
    For i = 0 To count
        Write #filenum, PictureData(i).DatName, PictureData(i).PicName

    Next i
    Close #filenum
End Sub
Sub LoadProjectPic()
    Dim filenum As Integer
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\picdat.dat" For Input As filenum
    count = 0
    'record all data
    While Not EOF(filenum)
        Input #filenum, PictureData(count).DatName, PictureData(count).PicName
        count = count + 1
    Wend
    Close #filenum
    For i = 0 To count
        If PictureData(i).DatName = Main.dcprojects.Text Then
            ProjectPic = PictureData(i).PicName
            i = count
        End If
    Next i
    On Error Resume Next
    Main.PictureNav.Picture = LoadPicture(ProjectPic)
End Sub
Sub LoadFloorPic()
    Dim filenum As Integer
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\picdat.dat" For Input As filenum
    count = 0
    'record all data
    While Not EOF(filenum)
        Input #filenum, PictureData(count).DatName, PictureData(count).PicName
        count = count + 1
    Wend
    Close #filenum
    For i = 0 To count
        If PictureData(i).DatName = Main.dcfloors.Text Then
            FloorPic = PictureData(i).PicName
```

```
        i = count
      End If
    Next i
    Main.FloorPicture.Picture = LoadPicture(FloorPic)

End Sub
Sub LoadRoomPic()
    Dim filenum As Integer
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\picdat.dat" For Input As filenum
    count = 0
    'record all data
    While Not EOF(filenum)
      Input #filenum, PictureData(count).DatName, PictureData(count).PicName
      count = count + 1
    Wend
    Close #filenum
    For i = 0 To count
      If PictureData(i).DatName = Main.dcRooms.Text Then
        Roompic = PictureData(i).PicName
        i = count
      End If
    Next i
    Main.RoomPicture.Picture = LoadPicture(Roompic)

End Sub
Sub AddFloorBox()
    SX = Round(CurrentStartX / 10, 0)
    SY = Round(CurrentStartY / 10, 0)
    EX = Round(CurrentEndX / 10, 0)
    EY = Round(CurrentEndY / 10, 0)

    'get all data from dat file
    Dim filenum As Integer
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\posdat.dat" For Input As filenum
    count = 0
    'record all data from file
    While Not EOF(filenum)
```

```
    Input #filenum, PositionData(count).Posname, PositionData(count).PosSX,
PositionData(count).PosEX, PositionData(count).PosSY, PositionData(count).PosEY,
PositionData(count).PosPjName, PositionData(count).PosPjPic
    count = count + 1
  Wend
  Close #filenum

  'Verify existing data
  For i = 0 To count
    If PositionData(i).Posname = PositionName Or i = count Then
      PositionRepeat = i
      i = count
    End If
  Next i
  PositionData(PositionRepeat).Posname = PositionName
  PositionData(PositionRepeat).PosSX = SX
  PositionData(PositionRepeat).PosEX = EX
  PositionData(PositionRepeat).PosSY = SY
  PositionData(PositionRepeat).PosEY = EY
  PositionData(PositionRepeat).PosPjName = PosProjectName
  PositionData(PositionRepeat).PosPjPic = PosProjectPic
  Close #filenum

  If PositionRepeat <> count Then count = count - 1
  'write data
  Open "c:\my thesis\posdat.dat" For Output As filenum
  For i = 0 To count
    Write #filenum, PositionData(i).Posname, PositionData(i).PosSX,
PositionData(i).PosEX, PositionData(i).PosSY, PositionData(i).PosEY,
PositionData(i).PosPjName, PositionData(i).PosPjPic

  Next i
  Close #filenum
End Sub
Sub AddRoomBox()
  SX = Round(CurrentStartX / 10, 0)
  SY = Round(CurrentStartY / 10, 0)
  EX = Round(CurrentEndX / 10, 0)
  EY = Round(CurrentEndY / 10, 0)

  'get all data from dat file
  Dim filenum As Integer
  Dim count As Integer
```

```
    filenum = FreeFile
    Open "c:\my thesis\posdat.dat" For Input As filenum
    count = 0
    'record all data from file
    While Not EOF(filenum)
        Input #filenum, PositionData(count).Posname, PositionData(count).PosSX,
PositionData(count).PosEX, PositionData(count).PosSY, PositionData(count).PosEY,
PositionData(count).PosPjName, PositionData(count).PosPjPic
        count = count + 1
    Wend
    Close #filenum

    'Verify existing data
    For i = 0 To count
        If PositionData(i).Posname = PositionName Or i = count Then
            PositionRepeat = i
            i = count
        End If
    Next i
    PositionData(PositionRepeat).Posname = PositionName
    PositionData(PositionRepeat).PosSX = SX
    PositionData(PositionRepeat).PosEX = EX
    PositionData(PositionRepeat).PosSY = SY
    PositionData(PositionRepeat).PosEY = EY
    PositionData(PositionRepeat).PosPjName = PosProjectName
    PositionData(PositionRepeat).PosPjPic = PosProjectPic
    Close #filenum

    If PositionRepeat <> count Then count = count - 1
    'write data
    Open "c:\my thesis\posdat.dat" For Output As filenum
    For i = 0 To count
        Write #filenum, PositionData(i).Posname, PositionData(i).PosSX,
PositionData(i).PosEX, PositionData(i).PosSY, PositionData(i).PosEY,
PositionData(i).PosPjName, PositionData(i).PosPjPic

    Next i
    Close #filenum
End Sub
Sub AddDataTypeBox()
    SX = Round(CurrentStartX / 10, 0)
    SY = Round(CurrentStartY / 10, 0)
    EX = Round(CurrentEndX / 10, 0)
    EY = Round(CurrentEndY / 10, 0)
```

```
    'get all data from dat file
    Dim filenum As Integer
    Dim count As Integer

    filenum = FreeFile
    Open "c:\my thesis\posdat.dat" For Input As filenum
    count = 0
    'record all data from file
    While Not EOF(filenum)
        Input #filenum, PositionData(count).Posname, PositionData(count).PosSX,
PositionData(count).PosEX, PositionData(count).PosSY, PositionData(count).PosEY,
PositionData(count).PosPjName, PositionData(count).PosPjPic
        count = count + 1
    Wend
    Close #filenum

    'Verify existing data
    For i = 0 To count
        If PositionData(i).Posname = PositionName Or i = count Then
            PositionRepeat = i
            i = count
        End If
    Next i
    PositionData(PositionRepeat).Posname = PositionName
    PositionData(PositionRepeat).PosSX = SX
    PositionData(PositionRepeat).PosEX = EX
    PositionData(PositionRepeat).PosSY = SY
    PositionData(PositionRepeat).PosEY = EY
    PositionData(PositionRepeat).PosPjName = PosProjectName
    PositionData(PositionRepeat).PosPjPic = PosProjectPic
    Close #filenum

    If PositionRepeat <> count Then count = count - 1
    'write data
    Open "c:\my thesis\posdat.dat" For Output As filenum
    For i = 0 To count
        Write #filenum, PositionData(i).Posname, PositionData(i).PosSX,
PositionData(i).PosEX, PositionData(i).PosSY, PositionData(i).PosEY,
PositionData(i).PosPjName, PositionData(i).PosPjPic

    Next i
    Close #filenum
End Sub
```

```
Sub FindFloorBox()
   CX = Round(CurrentEndX / 10, 0)
   CY = Round(CurrentEndY / 10, 0)
   PosProjectName = Main.dcprojects.Text
   PosProjectPic = ProjectPic
   'use for change data navigator
   Dim filenum As Integer
   Dim count As Integer

   filenum = FreeFile
   Open "c:\my thesis\posdat.dat" For Input As filenum
   count = 0
   'record all data
   While Not EOF(filenum)
      Input #filenum, PositionData(count).Posname, PositionData(count).PosSX,
PositionData(count).PosEX, PositionData(count).PosSY, PositionData(count).PosEY,
PositionData(count).PosPjName, PositionData(count).PosPjPic
      count = count + 1
   Wend
   Close #filenum

   'verify with database
   For i = 0 To count
      If PositionData(i).PosSX < CX And PositionData(i).PosEX > CX And
PositionData(i).PosSY < CY And PositionData(i).PosEY > CY And
PositionData(i).PosPjName = PosProjectName And PositionData(i).PosPjPic =
PosProjectPic Then
         CurrentFloor = PositionData(i).Posname
         i = count
      End If
   Next i
End Sub
Sub FindRoomBox()
   CX = Round(CurrentEndX / 10, 0)
   CY = Round(CurrentEndY / 10, 0)

   'use for change data navigator
   Dim filenum As Integer
   Dim count As Integer

   filenum = FreeFile
   Open "c:\my thesis\posdat.dat" For Input As filenum
   count = 0
   'record all data
```

```
    While Not EOF(filenum)
        Input #filenum, PositionData(count).Posname, PositionData(count).PosSX,
PositionData(count).PosEX, PositionData(count).PosSY, PositionData(count).PosEY,
PositionData(count).PosPjName, PositionData(count).PosPjPic
        count = count + 1
    Wend
    Close #filenum

    'verify with database
    For i = 0 To count
        If PositionData(i).PosSX < CX And PositionData(i).PosEX > CX And
PositionData(i).PosSY < CY And PositionData(i).PosEY > CY And
PositionData(i).PosPjName = PosProjectName And PositionData(i).PosPjPic =
PosProjectPic Then
            CurrentFloor = PositionData(i).Posname
            i = count
        End If
    Next i
End Sub
Sub InAddMode()
    AddMode = True
    FindMode = False
End Sub
Sub InFindMode()
    AddMode = False
    FindMode = True
End Sub
Sub InFloor()
    FloorShowed = True
    frmFloor.Show
End Sub
Sub InName()
    NameShowed = True
    frmName.Show
End Sub
Sub InMain()
    FloorShowed = False
    Main.Show
    frmFloor.Hide
    NewProject.Hide
    RoomShowed = False
    DataTypeShowed = False
End Sub
Sub InRoom()
```

```
    frmRoom.Show
    RoomShowed = True
End Sub
Sub InDataType()
    frmDataType.Show
    DataTypeShowed = True
End Sub
```

BIBLIOGRAPHY

Aitken P.G. (1998), *Visual Basic 6 Programming Blue Book*, Albany, NY: Coriolis

Binder, Stephen (1992), *Strategic Corporate Facilities Management*, New York, NY: Mcgraw Hill, Inc

Cotts, G.D. and Lee, M. (1992), "The Facility Management Handbook, New York, NY: Amacom

Fronckowiak, W.J. and Helda, J.D. (1999), *Visual Basic 6 Database Programming*, Foster City, CA: IDG Books Worldwide Inc

Gregory, H.M. (1988), *Facilities Maintenance Management*, Kingston, MA: RSMeans Company, Inc

Petroutsos, E. (1998), *Mastering Visual Basic 6*, San Francisco, CA: Sybex

Petroutsos, E. (2000), *Mastering Database Programming with Visual Basic 6*, San Francisco, CA: Sybex

RSMeans (1996), *Cost planning & Estimating for facilities maintenance.* Kingston, MA: R.S. Means Company, INC.

Shearer, K.A. (1922), *The Inventors' and Owners' Guide to Office Building Management*, Homewood, IL: Business One Irwin