WEB-ENABLED COST ESTIMATION FOR CASTINGS

Dissertation
Submitted in partial fulfillment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY

By

MAHESH M. NAIR
Roll No. 00310015

Under the guidance of
Prof. B. Ravi

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, BOMBAY
JANUARY 2002

Table of Contents

Abstract

Casting is a major manufacturing process, which requires an understanding of a wide range of geometrical, material and process parameters. In today's market of intense competition, where the need of the hour is to develop quality products at low cost and short lead time, innovative and cost effective designs are the means to success. Further, with global manufacturing trends, the concept of concurrent engineering and web-based collaborative engineering are fast gaining pace. Early cost estimation, necessary for design-to-cost philosophy, serves as a decision making tool for the designer when a choice is to be made from a range of alternatives. In this project, a web enabled cost estimation software for cast products has been developed to demonstrate the feasibility of early cost estimation, useful for product design team members. The program requires only a web browser and does not require any special skills. It has been developed after a study of the various casting processes and their process parameters affecting the cost of the final product. Based on the study a cost model was developed as a function of product requirements. To accurately estimate the tooling cost, perhaps the most important component, a regression analysis based on actual industrial data was conducted. The cost model was successfully implemented and tested over the Internet. It is expected to cater to the needs of even those practicing engineering, who may be located in remote areas.

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

1.1 Casting Industry

Metal casting is one of the oldest industries in human society. The art of foundry is as ancient as the dawn of civilization. Even the craftsmen of the Greek, Roman and Indus valley civilizations had practiced the art of foundry.

A casting may be defined as a "metal object obtained by allowing molten metal to solidify in a mold", the shape of the object being determined by the shape of the mold cavity. Certain inherent advantages of the metal casting process, which cause it to be selected over the other processes, are listed below [Heine, 1955].

1. The most intricate of shapes, both external and internal, may be cast.
2. Because of their metallurgical nature, some metals can only be cast to shape since they cannot be hot-worked into bars, rods, plates or other shapes from ingot form as a preliminary to other processing.
3. Objects may be cast in a single piece that would otherwise require construction in several pieces and subsequent assembly.
4. Metal casting process is highly adaptable to the requirements of mass production.
5. Extremely large, heavy metal objects may be cast when they would be difficult or economically impossible to produce otherwise.
6. Some engineering properties are obtained more favorably in cast metals, like machinability, uniform properties from a directional standpoint, strength and lightness in certain light metal alloys that can be produced only as castings and good bearing. In general, a wide range of alloy composition and properties is produced in cast form.
7. A decided economic advantage exists as a result of any one or a combination of the above points.

Castings have a wide and extensive range of applications. They range in size from a few grams to more than 100 tons. In function, they range from decorators' trinkets to critical bridge structure parts [Heine, 1955]. There are around 6000 foundries in India, producing more than 3 million tones of casting [Vasvani, 1997]. Table 1. gives the statistics of various kinds of cast products manufactured in India.

Table 1.1 Various Types of Cast Products Manufactured in India [Vasvani, 1997]

Castings	Weight Poured in Million Tones
Gray Iron	2.360
Steel	0.400
Malleable S. G.	0.268
Non-Ferrous	0.020
Total	3.048

1.1.1 Major Casting Processes

Of the various casting processes the important ones, that constitute the bulk of the castings produced, are sand casting, die-casting processes and investment casting. A brief overview of these processes has been shown in the following tables.

Table 1.2 Sand casting process [Nee, 1999]

Process	Schematic Diagram	Description
Sand casting Pattern making Molding Metal melting Metal pouring Solidification Mold braking Fettling		**Material** Cast Iron, steel, Aluminum, Copper alloys **Accuracy** ± 0.5 to 1.5mm **Mold Material** Foundry sand (silica +clay + additives) **Mold life** 1 casting/mold

Table 1.3 Pressure die casting [Nee, 1999]

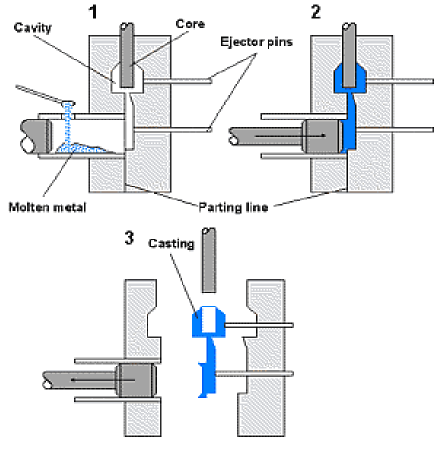
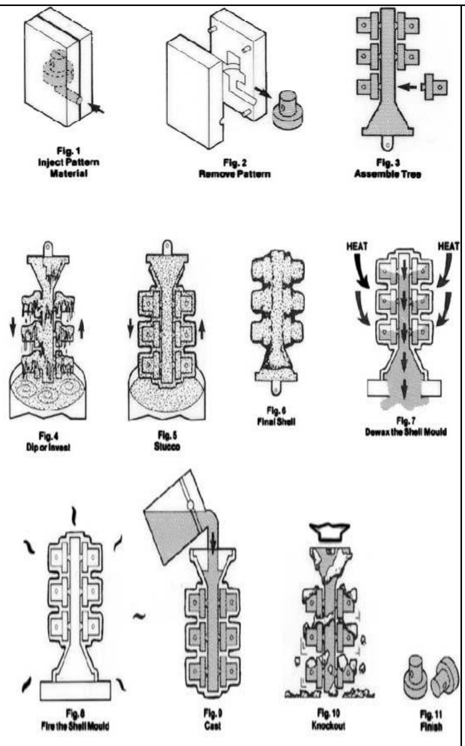
Pressure Die casting Mold closing Shot sleeve filling Metal injection Solidification Mold opening Ejection Fettling		**Material** Al, Zn, Cu, Mg alloys **Accuracy** ± 0.02 to 0.05mm **Mold Material** H 11 / H 13 **Mold life** 0.5 to 1million shots

Table 1.4 Investment casting process [Nee, 1999]

Investment casting Wax mold making Wax pattern making Tree building Ceramic shell preparation De-waxing Metal pouring Solidification Mold breaking Fettling		**Material** Aluminum, steel, Zinc, Copper alloys **Accuracy** ± 0.2 to 0.5 mm **Mold Material** ***Wax injection mold***; Al, MS, OHNS, EN24 ***Casting***: Ceramic **Mold life** ***Wax injection mold***: 2000 –10 million patterns ***Casting***: 1 casting/mold

1.1.2 Process Flow Diagram

The flow process diagram gives an overall picture of the casting process. The following figure shows the process flow for sand casting process.

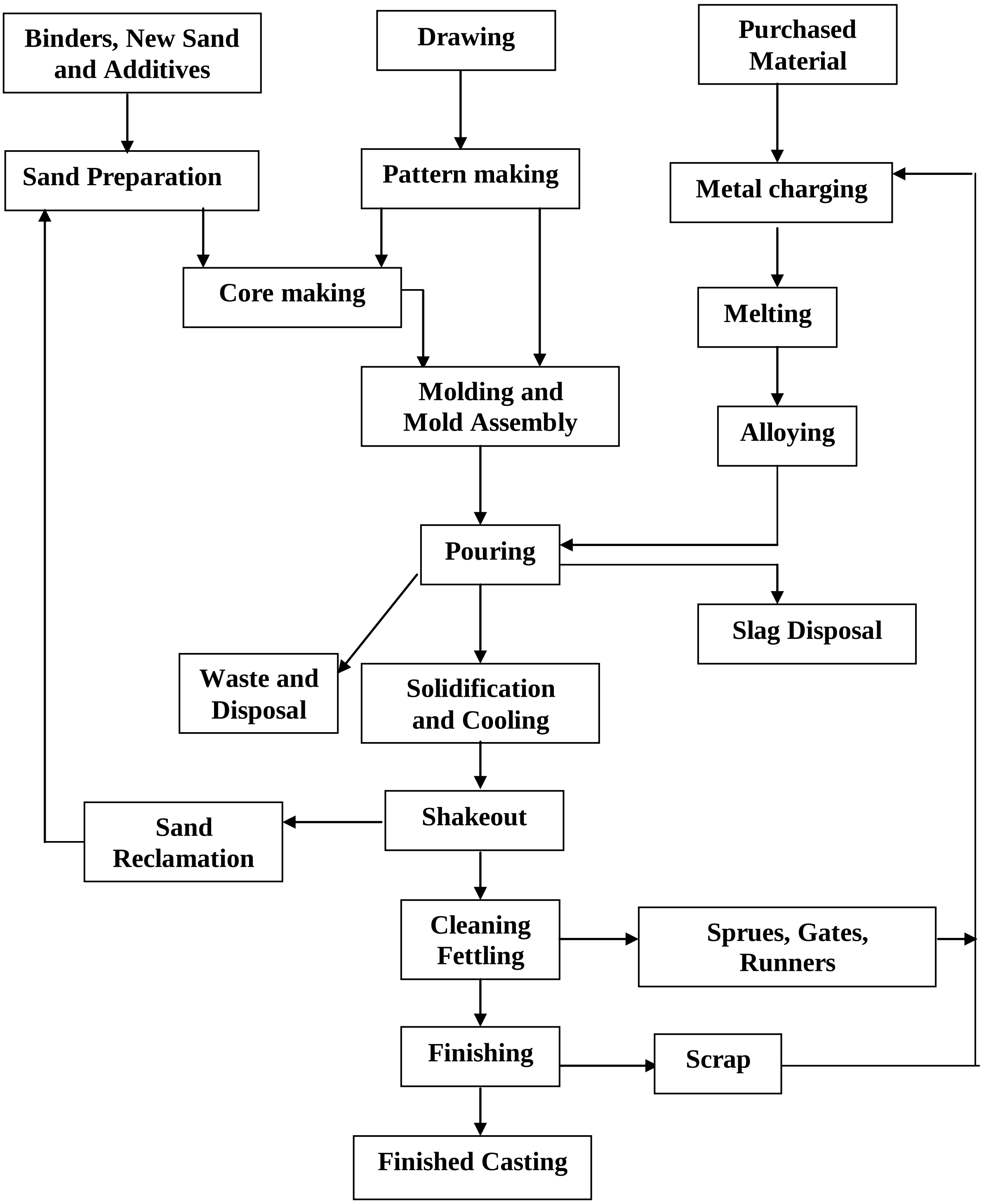


Figure 1.1 Process flow diagram of sand casting [Creese, 1988]

The first step is preparing the pattern based on the part drawing. At the same time, the sand mixture is being prepared. Simultaneously, the metal of required composition is melted using a furnace. Once the cores are prepared, the molding takes place using the molding machines that maybe available in the shop. Then with the gating system in place the molten metal at appropriate temperature is poured into the mold cavity. The metal is allowed to solidify and cool. Then the mold is broke open and the part is removed. The gating, risers, sprues, etc. are chopped off and reused as scrap. The casting is cleaned of the sand and the finishing operations are performed. A typical flow process diagram of a sand casting foundry is shown in figure 1.1 [Creese, 1988].

1.1.3 Current Challenges

Although the process appears to be quite simple involving no assembly operations, casting is a highly complex and challenging task, requiring an understanding of a large number of geometric, material and process parameters involved [Swift, 1997].

Figure 1.2 Internal and external pressures on casting industry [Swift, 1997]

Pressures, both external and internal, further aggravate these engineering challenges faced by the foundries today as shown in figure 1.6 [Swift, 1997]. External pressures include those imposed by the customer, competition, suppliers and regulatory bodies whereas internal pressures relates to human resources, technology, marketing strategy and management practices. There is a significant gap in the level of technology deployed in casting sector compared to others which is making it difficult to integrate casting process with other processes for the manufacturing and assembly of engineering products. One solution to bridge this gap is the use of software tools which provides efficient decision making capability to product designers, tool designers, foundry managers and material suppliers and also provides effective means of communication between them.

1.2 Product Design and Cost

Product design and development is the most important and crucial stage in the product life cycle. At this stage, certain decisions are to be made based on the evaluation and comparison of the various alternative solutions available for product design, and the most suitable alternative is to be selected for further development [Leibers, 1997]. The manufacturing costs, lead time and quality are the most important parameters in this decision making process. While the quality of the product can be controlled in the subsequent stages, it is almost impossible to control the cost of the product if changes are to be made in the product design at the later stages in the product development cycle.

Figure 1.3 Cost and impact of various activities [Hundal, 1993]

The reason for the above being the fact that up to 85% of the products' cost is committed through the decisions made at the product design stage, although only 5% of the total developmental costs are actually incurred [Hundal, 1993].

In addition to this, a set of dominant trends in the business environment that have influenced the competitiveness of the companies are shortened product life-cycle, increasing diversity, variety and complexity of products and the customers becoming increasingly more sophisticated and demanding customized products more closely targeted to their needs. This has led to pressures of continuous product improvements, leading to ever increasing functionalities and features. Figure 1.8 shows the trend between product life times and their development times over a period of time [Chanan, 1994]. The figure clearly indicates that if the product development times are not reduced significantly, it will cause delay in bringing the product to the market resulting greater losses of profit. Thus, profit comes from early product launch.

Figure 1.4 Product development time versus the product's lifetime [Chanan, 1994]

In order to be successful, a product needs to be both, innovative in design and cost effective. The concept of survival triplet and survival zone help decide the commercial viability of a product based on the values of its three attributes viz., cost/price, quality and functionality [Cooper, 1997]. Only products with values along each of these three dimensions that are acceptable to the customer have a chance of being successful. To identify a product's survival zone the survival range for each characteristic in the survival triplet must be determined. The survival range is defined by determining the minimum and maximum values that each characteristic can have, for the product to be successful.

The survival zone is the volume created by connecting the three minimum value and the three maximum values [Cooper, 1997].

Figure 1.5 The survival triplet of a product [Cooper, 1997]

Figure 1.6 The survival zone of a product [Cooper, 1997]

1.3 Concurrent Engineering

Concurrent Engineering (CE) is the philosophy for improving design communication and aiding the process of recognizing and resolving traditional conflict [Creese, 1990]. As shown in the figure 1.7, it helps in parallel processing of the various manufacturing activities thereby reducing lead time and cost and improving on quality. It leads to improvements in many areas such as communication, quality, production processes, cash flow and profitability as compared to the Sequential Engineering where activities follow each other in occurrence leading to greater costs and lead times [Ravi, 1996]. Computer-based decision support systems would go a long way in implementing the CE philosophy in the organizations. Typical computer-based tools include engineering data management (EDM) tools, modeling and simulation (CAD/CAM) tools, scheduling tools and costing tools [Chanan, 1994].

Figure 1.7 Information flow in traditional versus concurrent engineering [Creese, 1990]

1.4 Web-Based Collaborative Engineering

In the past, most product design activities were performed in centralized engineering departments, often with people seated side-by-side working on the same projects. Communication was direct, and problems could be resolved quickly as face-to-face interaction was possible.

Today, many organizations have distributed design and production facilities around the world, and current methods for communication are unsuitable. Face-to-face meetings necessitate costly and time-consuming travel; telephone conversations are inadequate for describing changes to drawings and design models; faxed drawings are often illegible and the size of e-mail attachments is limited. Video conferencing requires high bandwidth and does not allow manipulation of CAD data. The FTP (File Transfer Protocol) utilities can be used to exchange product drawings, but team members must run CAD software independently to view and manipulate design models. These barriers have led to the development of a new web-based paradigm for communication and sharing of information among globally distributed team members.

1.5 Problem Definition

Product designers often design components with limited knowledge of manufacturing processes. The result is that the tooling is unnecessarily expensive or the process is not fully utilized. Also, when a number of alternative solutions are available, there needs to be some criteria based on which the selection of the most suitable process is to be made. Further, the implementation of the design-to-cost principles and the concurrent engineering philosophy becomes possible with the help of software tools like cost estimation which could prove to be very handy to give a quick estimate of the product in the early design stages, thereby making the design changes inexpensive and also helps in determining the most suitable alternative to manufacture the product.

Thus, the objective of this project is to develop a methodology for early cost estimation of cast products that would be useful for product designers, and implement the same in a web-based framework for facilitating collaborative engineering.

1.5.1 Objectives

The objectives of this project have been listed down.

- To identify the various factors affecting the casting process and establish their relationship with product cost.
- To develop an integrated cost model involving the various factors that have been identified, to determine the individual cost components and thereby the total cost of the product.
- To implement the cost model developed as cost estimation software and web-enable the same to make it available to a wide range of users irrespective of their location.

1.5.2 Scope and Approach

The scope of the project has been limited to the main casting processes viz., sand casting, die-casting and investment casting processes. The approach that has been adopted has been described in brief below.

- An in-depth study of the various casting processes is conducted. The steps involved the processes their important parameters were studied and analyzed.
- The factors that affected the processes profoundly and had an influence over the cost of the product were identified and their relation with the cost of the product was analyzed.
- The design-to-cost and concurrent philosophies were studied and the relevance of cost estimation in their successful implementation was established.
- The structure of cost estimation of model, the need for cost estimation and the various cost estimation techniques were studied and analyzed.
- An integrated cost model, involving the important factors identified, was developed wherein the individual cost components are calculated to give the total cost of the product.
- The cost model so developed has been codified into software and implemented over the internet so as to cater to the needs of wide range of users.

1.6 Thesis Organization

In the **first** chapter, an overview of the project is given. This chapter talks about the major casting processes, the significance of cost estimation at product design stage, the concurrent engineering philosophy, collaborative engineering over the web and finally explains the objectives of the project. The **second** chapter reviews the technical literature related to design-to-cost (fundamentals, implications and implementation), the concept of cost estimation and various cost estimation techniques, DFM guidelines for cost reduction and finally about web-based applications for castings. The **third** chapter describes the system design with reference to a casting data modeling and web-based framework, and describes the program flow chart. The **fourth** chapter explains the cost model developed along with the various modules within the same and finally explains the regression analysis performed for estimating the tooling cost. The **fifth** chapter deals with the implementation and testing of the web-based casting cost estimation program and describes a sample session. The **last** chapter states the contribution, limitations and scope for future work of the project.

CHAPTER 2

LITERATURE REVIEW

Innovative and cost effective design being the means to success in today's competitive market, the need is to focus on methods to achieve the same. The design-to-cost philosophy, wherein a target cost is determined before developing the product is a step in the right direction. Design to cost is the iterative redesign of a project until the content of the project meets a given budget. Effective product cost management requires a design to cost philosophy as its basis since a substantial portion of the product's cost is dictated by decisions regarding its design. Design to cost is a management strategy and supporting methodologies to achieve an affordable product by treating target cost as an independent design parameter that needs to be achieved during the development of a product [Dean, 1991]. Further, to determine the target cost and then compare it with the product cost, an accurate cost estimation technology is required. Also, in today's paradigm of concurrent engineering wherein the product development team-members may be situated at geographically distant locations, collaborative engineering over the web is the order of the day.

The design-to-cost approach involves three steps. The first step is setting up of the target cost of the product. Next step involves the cost estimation of the product based on its design features and in the last step, the product is redesigned making use of the cost reduction techniques so as to meet the target cost. In the next section, we deal with the target costing methodology, whereas the cost estimation and cost reduction techniques will be dealt in the following sections.

2.1 Target Costing

A number of product designs are often not successful when implemented, though they are backed by a strong market research in terms of customer needs. Simply the product is not accepted by customers despite it has all attributes of good design because it is not affordable to customers. Everyday customers buy products with functions, features and performance in excess of their need and how much money is wasted on these unneeded capabilities. Target costing aims to reduce these unneeded capabilities and thereby reduce the cost of the product, thus increasing its economic viability.

2.1.1 Definition

"Target costing is structured approach to determine the life cycle cost at which a proposed product with specified functionality and quality must be produced to generate the desired level of profitability over its life cycle when sold at its anticipated selling price [Cooper, 1996]."

2.1.2 Structure of Target Costing

The target costing system relies heavily on the cardinal rule: the target cost of product can never be exceeded. Without this rule, target cost system typically lose their effectiveness. The primary objective of the cardinal rule is to stop the steady and relentless creep in the product functionality and cost that occurs when product designers say, "if we add just this feature, the product will be so much better [Cooper, 1997]." The target costing process can be broken into three major sections as depicted in figure 2.1.

Figure 2.1 The target costing process [Cooper, 1997]

Market Driven Costing: Target costing begins with an analysis of the market conditions. The aim of this exercise is to identify the allowable cost of each product. This would be the cost at which the product must be manufactured, if it is to earn its target profit margin at its expected target selling price.

It involves setting of the long-term sales and profit objectives of the firm and using target costing as a means for achieving the same. The next step is the structuring of the product line of the firm to achieve maximum profitability, by evaluating the products based on their survival zone. Once a product's survival zone has been located, next step involves the determination of its target selling price. Having fixed the target selling price of the product, its target cost has to be fixed based on the target profit margin that the firm must achieve to fulfill its long-term objectives.

Product-Level Target costing: Having established the target cost of the product based on market-driven costing, the next exercise is to set the product-level target cost. It then focuses on the creativity of the designer to design the product within the target cost. This is an iterative process and various cost reduction tools are used by the designer to reduce the product cost, till it matches with the target cost.

Component-Level Target Costing: In this stage the product-level target cost is further broken down to component-level. Here the aim is to identify how much the firm is willing to pay for the individual components, which it procures from vendors. The goal here is to discipline and focus the creativity of the suppliers in ways beneficial to the firm.

2.2 Cost Estimation methods

Cost estimation is the determination of costs for a production situation that is likely to be encountered. It is a prediction of product costs and is one of the main inputs for economic evaluation. Cost estimation of any product is a complex process and requires engineering skills, as it involves understanding the following items [Creese, 1988]:

1. The product design features and the impact of these features on its cost.
2. The suitability of alternative materials, which will satisfy the functional requirements and the processing methods.
3. The alternative technologies available for processing the material of concern and the cost of each process.
4. The costs involved in ensuring the quality of the product in all stages of production.
5. The costs of servicing the product during its usage.
6. The implications and costs involved in product liability suits, in case of product failure.
7. The environmental effects and costs of product disposal at the end of the product life.

An understanding of these items requires knowledge of the principles of engineering, materials, manufacturing processing, inspection methods, quality control, repair and servicing, product safety, and environmental safety as well as the principles of costing.

2.2.1 Components of Cost Estimation

The typical components of a cost estimation are [Creese, 1988]:

a) *Design Cost:* It is estimated from expected time for the design of that component. This may be done based on similar jobs previously done, but for new or complicated jobs, it is essential to consult the designer. If the design were done by outside contractor, the amount paid to him would be the design cost.

b) *Engineering/Drafting Cost:* It includes the engineering time in simulations and modeling to validate the design and to prepare the engineering drawings needed for production. This would be primarily engineering man-hours and computer time costs.

c) *Research and Development Cost:* It includes costs of developments and prototypes. These costs are difficult to estimate, especially for new products, and often are included in overhead costs. The developmental and prototype costs can often be attributed to a particular product during its development, but it is difficult to estimate these costs prior to the start of development.

d) *Material Cost:* The cost of materials can be estimated from the design by determining the types, amount and shapes required in production. Allowances for wastage, spoilage and scrap should also be included in the cost estimates.

e) *Labor Cost:* The cost for labor includes the direct labor time and rate for the product. The determination of these values requires a thorough knowledge of the operations performed, sequence of operations and machines and tools used. Some of these costs are included in overheads.

f) *Quality Cost:* The quality cost includes the cost of inspection, the cost of inspection equipment, the cost of maintaining process control charts and costs associated with improving product quality.

g) *Tooling Cost:* The tooling cost includes the tool costs, cost of fixtures, the cost of sharpening tools, tool storage costs, etc. Cutting fluids costs may be included in the material costs, tooling costs or as a part of overhead costs.

h) *Overhead Cost:* The overhead costs mainly include equipment depreciation overhead, inventories, material storage, etc. All these cost components cannot be directly charge to a specific product.

2.2.2 Typical Cost Estimation Model

A typical cost estimation model is shown in figure 2.2. It is a five-block structure [Liebers, 1997]. The main task of costing data generation for product design is out in the central block. The cost control information is provided from the topside. It comprises of the initiation of the cost estimation process, together with the estimation requirements like accuracy, personnel, available times, etc. On the left-hand side, the input information is supplied which describes the situation for which the cost estimation is to be done. This information includes a full description of the regarded alternative solutions for which the estimate has been prepared.

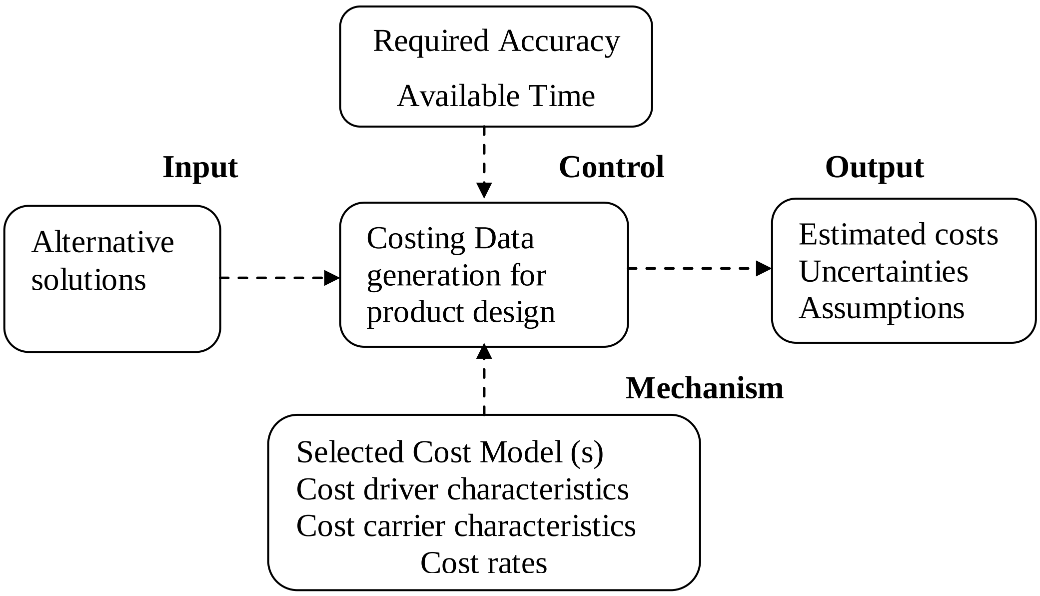


Figure 2.2 Typical cost estimation model [Liebers, 1997]

On the bottom side, the so-called 'mechanism' is supplied. The mechanism represents the means by which the input is transformed into output. In case of developing cost estimation for design decisions, the mechanism consists of selected cost models that suit the specific situation, regarding the input and control information. Finally, the right-hand side block represents output, which are estimated costs, uncertainties and assumptions.

2.2.3 Traditional Technique

Traditional Cost estimation technique, also known as orthodox cost accounting, is used in foundry industry since many years [Creese, 1988]. In this method of cost estimation, the individual cost components are expressed in a stepwise fashion referred to as 'ladder of costs'. The sum of direct material, direct labor, direct engineering and direct expenses is the 'Prime Cost'. Factory expenses when added to the prime cost give the 'Factory Cost'. In this way, by adding subsequent costs, 'Production Cost', 'Total Cost' and finally the 'Selling Price' can be arrived at. The relationship between the various cost components is shown in the figure 2.3 and are expressed by the following equations:

Prime Cost = Direct Material Cost + Direct Labor Cost + Direct Engineering Cost + Direct Expenses.

Factory Cost = Prime Cost + Factory Overheads.

Figure 2.3 Ladder diagram of traditional cost estimation [Creese, 1988]

Production Cost = Factory Cost + Administration Expenses.

Total Cost = Production Cost + Marketing, Selling and Distribution Expenses

The traditional technique has a few drawbacks. While the technique works well under unchanging conditions of same product mix for longer periods, it increasingly leads to distortions under complex operations. Another important shortcoming of the traditional technique is the allocation of the overheads amongst the products. Overheads are distributed in the same ratio as the respective costs of direct labor in those products. However, the manufacturer of the two products normally uses the overheads in the proportion that bears no relationship with the labor costs [Majumdar, 1997]. Moreover, with the advent of producing widely varied product lines, within the same facility, the reduction of labor costs due to automation and the increase in sales and distribution costs, the traditional cost estimation technique now proves to be inefficient.

2.2.4 Activity Based Costing Technique (ABC)

Activity based costing, as the name suggests, involves costing a product by adding up the costs of various activities involved in its manufacturing. The technique furnishes more accurate cost information for strategic and management decisions [Ong, 1993]. As compared to the traditional cost estimation technique, ABC is more effective in terms of allocation of the overhead costs amongst various products. Also, in the present scenario of producing wide range of products with lot of raw materials handling using a variety of processes, using different product lines and addressing several different customer segments, ABC is a more accurate cost estimation tool. The big advantage of ABC is that it is capable of providing managers with fairly accurate cost information for making wide range of decisions. ABC also provides information about activities for continuous improvement, including the resources required to perform activities and distinction between value added and non-value added activities. The most important benefit that ABC offers is the gain that comes in the form of enterprise-wide-cost and process-management opportunities [Majumdar, 1997]. However, the ABC technique has one important drawback. Although it gives better-cost estimates, the technique however, has one fundamental barrier of the proportion of overhead costs to total costs. If the direct-to-total cost ratio of the company is high, even traditional technique can give better cost estimation and using ABC for such an application may not be even suitable as lot of efforts and time are required to be put up for estimating cost using this technique. However, where the ratio is quite low, ABC can be used profitably. Steps involved in implementing ABC technique are shown in figure 2.4 [Bharara, 1996]. These are explained below:

1. *Identify the activities*: The concept of ABC is based on splitting up the life cycle of a product into various functions or processes or activities. To better understand the structure and operations of the organization, significant activities carried out should be identified. For example, a foundry process can be split into activities such as core making, molding, melting, etc. that forms the basic tasks required for manufacturing of the products.
2. *Develop a cost flow model*: For developing a cost flow model for the product, it is essential to identify the basic cost pools for a product as it helps in determining the types of costs flowing into various activities at various steps. The cost pools are the various types of costs incurred in the manufacturing of the product and mainly constitute material, energy, labor, equipment depreciation or overheads, premium, etc. Once identified, the cost pools are associated with the activities. This is a two-step procedure. In the first step, the physical change that the activity causes to the product should be identified. The second step involves further analyzing the activities into various tasks, and associating the cost with each task depending on the cost drivers.
3. *Determine the activity drivers and driver quantities*: An activity cost driver is a measurable factor that is used to assign cost between various activities and from one activity to another [Majumdar, 1997]. This is the most important step in implementing the ABC system. The cost driver directly reflects the amount of cost associated with that particular activity. For example, cost driver for machine setup activity is the amount of time taken which multiplied by the machine hour rate gives the set up overheads. Similarly, number of units, volumes of products, etc. are the cost drivers.
4. *Collect cost-driver data*: This is the most important step in implementation. The ABC technique helps to identify the major cost-carrying areas in a production process as it splits the process into various activities. However, for calculating the activity costs, it is essential to collect the related cost-driver data and store it in a proper format, which can be accessed by the product designers when and where required. Main source of the cost-driver data is the cost accounting system [Sheldon, 1993]. It usually consists of a number of departmental ledgers, each recording the consumption and transaction of internal resources. For instance, the personnel department keeps personnel record regarding wages. The sales and marketing department is apparently responsible for pricing products, dealing with market demand, customer analysis, etc.

Figure 2.4 Steps involved in implementing ABC technique [Ong, 1993]

5. *Calculate activity costs*: Next step is to calculate the activity costs. The main inputs required are the amount of cost drivers and the related cost rates. In addition to these factors, the activity cost depends on the following.

Cost basis: The cost incurred are based either an individual part, known as per unit basis or on the weight of the part, known as per kg basis. The choice of the cost basis depends upon the type of products produced in the organization.

Activity hierarchy: It is the classification of activities into various types depending upon their level [Wen-Hsien, 1996]. There are several types of activities, which are as follows:

Product Development activities: Activities related to the development of a very new product. For example, R&D expenditure.

Unit-level activities: Activities related to the production of a unit of product. It varies with the volume of output. E.g. direct material cost.

Batch-level activities: Activities which assume that the inputs vary in proportion to the number of batches produced, but the costs are amortized for all units produced in that batch. For example, mold sand preparation.

Product-level activities: Activities necessary to support the production of each different type of product. For example, injection mold dies.

Facility-level activities: Activities that support the facility's general manufacturing. For example, depreciation cost.

6. *Product Cost*: Finally, the product costs are determined by adding up the corresponding activity costs.

2.2.5 Parametric Costing Technique

Parametric estimating, is a technique that uses validated relations between a project's known technical, programmatic and cost characteristics and known historical resources consumed during the development, manufacture, and/or modification of an end item [NASA, 1999]. A number of parametric techniques exist that can be used by practitioners to estimate costs. These techniques include cost estimating relationships (CERs) and parametric models. CERs are defined as mathematical expressions or formulas that are used to estimate the cost of an item or activity as a function of one or more relevant independent variables, also known as cost driver(s).

Parametric models are more complex than CERs, because they incorporate many equations, ground rules, assumptions, logic, and variables that describe and define the particular situation being studied and estimated. Parametric models make extensive use of databases by cataloging program, technical and cost history. However, till recently the use of parametric estimating techniques for purpose of cost estimation has been limited due to following reasons:

- Cultural resistance, because many people in acquisition of community expressed greater comfort with the traditional techniques of estimation.
- Limited availability of guidance on how to prepare, evaluate an alternative using parametric estimating techniques.

Cost estimation relationships (CERs) are used by a large number of companies to develop estimation models. CERs are mathematical expressions of varying degree of complexity, expressing cost as a function of one or more cost driving variables [NASA, 1999]. The relationship may utilize cost-to-cost variables, such as manufacturing hours to quality assurance hours, or cost-to-noncost variables, such as engineering hours to number of engineering drawings. The continuum of CERs is synonymous with the term parametric estimating methods. Parametric estimating methods are defined as estimating techniques that rely on theoretical, known or proven relationship between item characteristics and the associated item cost. The estimating relationship can range in complexity from something very simple; such as a numerical expression of a value or a ratio (typically expressed as a percentage) to something very complex; such as a multi-variable mathematical expression. As the complexity increases the set of expressions tend to be termed as a cost model. A cost model is a series of equations, ground rules, assumptions, relationships, constants, and variables that describe and define the situation or the condition being studied.

2.3 Cost Reduction Methods

In today's turbulent market environment, companies are compelled to rethink sequential planning procedure with the objective to integrate and synchronize the product design, process planning and the cost estimating activities. Proliferated methodologies referred to by the synonyms Simultaneous Engineering and Concurrent Engineering strive for reducing the lead time to market, trimming development and manufacturing costs, and enhancing the quality of products in the overarching sense of Total Quality Management. There is increasing pressure on firms to innovate and market newer products in order to sustain the fierce competition. However, developing new technology or product alone is no guarantee for success, for when a product succeeds it is more due to economic reasons than technical ones [Eias, 1998]. Therefore, the need of the hour is of techniques that would help reduce cost of the product without compromising on its quality. Value Engineering (VE) is one such technique, which helps fulfill the above-mentioned goals.

2.3.1 Value Engineering

The society of American value engineering defines value engineering as " the systematic application of recognized techniques which identify the function of a product or service, establish a monetary value for that function, and provide the necessary function reliably at the lowest overall cost [Eias, 1998]".

Different customers will interpret the value of product in different ways. Its common characteristic is a high level of performance, capability, emotional appeal, style, etc. relative to cost.

VE however is concerned only with economic value, which is defined as:

"The lowest cost to reliably provide the required function or service at the desired time and place and with the essential quality" [Eias, 1998].

The scope of the VE effort depends on the size and complexity of the project. However, the return on investment is highest when VE is performed in the early stages of the project life cycle, when implementation costs are lower. That is the time before major decisions have been incorporated into the design and when VE recommendations have the greatest impact on costs. In general, a project goes through five stages of development. The following figure shows the five stages and their sequence in product development cycle.

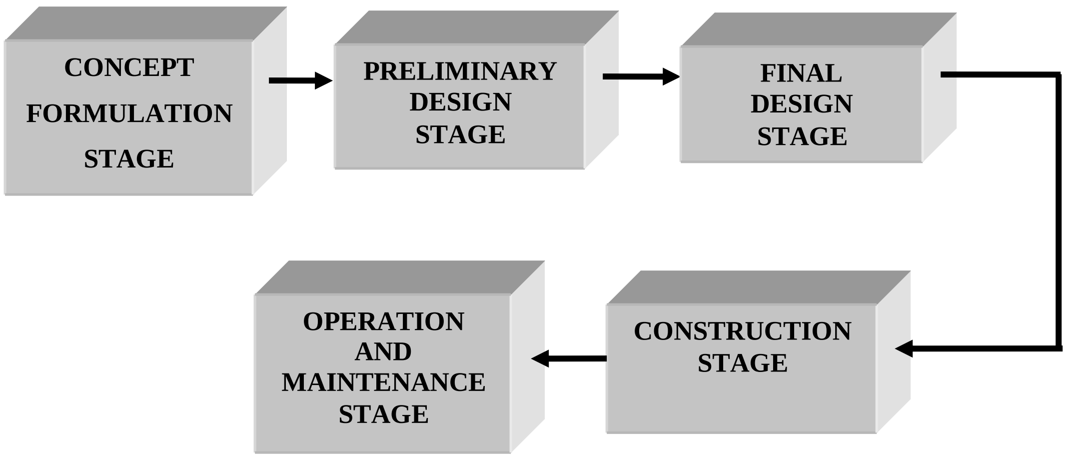


Figure 2.5 Stages of a project

VE, when applied to each of these stages, is aimed at different specific elements of the project development.

a) *The Concept Formulation/Performance Specifications*

The purpose of the concept formulation phase is to translate general requirements of the customer into performance specifications. In most cases, the final project picture is not yet known, meaning that decisions are made in somewhat unspecified conditions. VE efforts in this phase are directed towards furnishing inputs that will achieve the functions sought, at the lowest possible cost. Improvements generated during this phase produce benefits that last throughout the life of the project.

b) *Preliminary Design Stage*

During the preliminary phase, approved concepts are defined, and design specifications are started. Sufficient detailed information is developed to substantiate all quantities and costs that have been presented in the program directive. This is a good time to question performance characteristics and revise them if necessary. A VE study that analyzes requirements, technical characteristics, and the design tasks may reveal possible alternatives, offering improved value.

c) *Final Design Stage*

It is during the final design phase that design specification details are formulated and schedules created. VE efforts in this phase are usually limited to eliminating unnecessarily restrictive details. Usually redesign at this stage cannot be economically accomplished, unless the life cycle savings potential are large enough to justify the expense.

d) *Construction Stage*

During the construction phase, VE is accomplished by reviewing specific contract requirements and initiating change orders. Since change orders tend to increase contract cost, they should be subjected to value analysis to prevent adding nonessential functions and to facilitate finding other solutions that would lower the cost.

e) *Operation and Maintenance Stage*

The total cost of ownership is affected by operation and maintenance costs. Reducing these costs, results in lower life-cycle cost. VE studies during this phase offer an opportunity to make changes that were not made earlier. VE studies during this phase could result in cost savings by extending the life of an item through the use of new materials, processes and/or design, reduced repair costs, savings in energy and other operating costs.

Another technique, which acts as a useful tool in cost reduction, is the DFM guidelines. The next section deals with the DFM guidelines in general and those specific to casting.

Figure 2.6 Life Cycle phases and savings potential [Eias, 1998]

2.3.2 Design for Manufacturability (DFM)

DFM is recognized as a key to simultaneously minimizing the manufacturing cost, assuring the product quality and realizing the productivity increase promised by advanced manufacturing technology. The first DFM software introduced in mid 1980, however its application is getting much importance from 1995 onwards only. DFM guidelines help the designer to design the product for low cost.

However, the application of DFM must consider the overall design economics. It must balance the effort and cost associated with development and refinement of the design to cost and quality leverage that can be achieved. In other words, greater effort to optimize a products design can be justified with higher value or higher volume products. Design effectiveness is improved and integration facilitates when [Crow, 2001]:

- Fewer active parts are utilized through standardization, simplification and group technology, retrieval of information related to existing or preferred products and process.
- Producibility is improved through incorporation of DFM practices.
- Design alternatives are evaluated and design tools are used to develop a more mature and producible design before release for production.
- Product and process design includes a framework to balance product quality with design effort and product robustness.

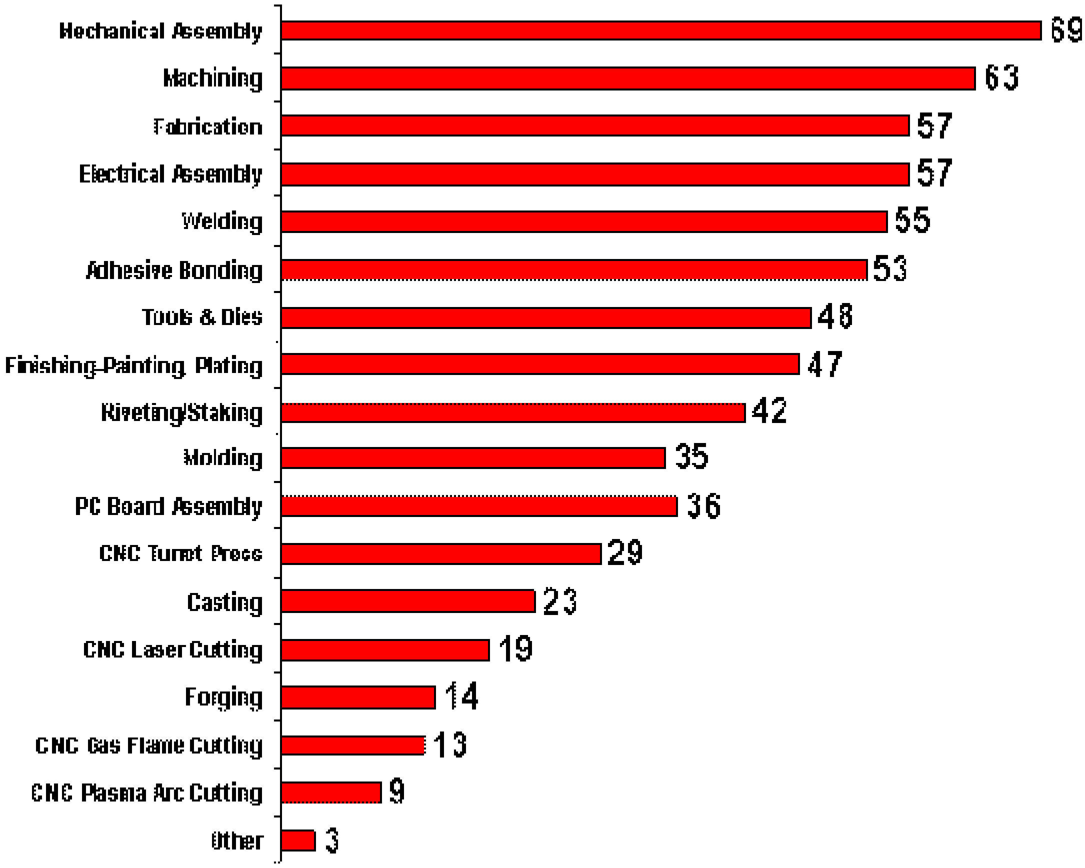


Figure 2.7 DFM users distribution in 1998 survey of leading design engineers [Maryland, 2001]

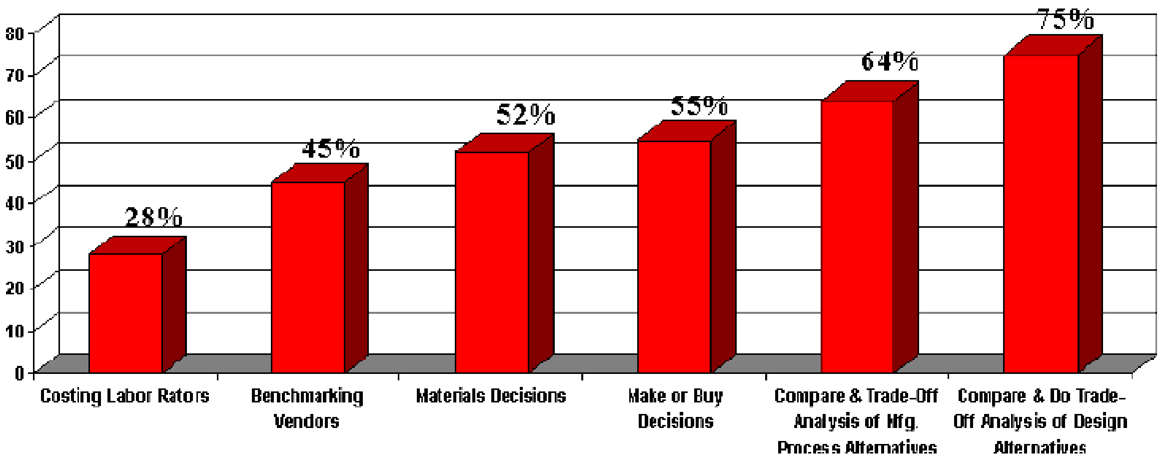


Figure 2.8 Purpose of using DFM in product development [Maryland, 2001]

The above figure shows the various purposes for which DFM is used during product development.

A number of guidelines have been formulated to help design a product for manufacturability. These guidelines help the designer in designing the products that are easy to manufacture. Some of the main guidelines have been listed below [Crow, 2001; Bralla, 1988]:

- Reduce the number of parts to minimize the opportunity for a defective part or an assembly error, to decrease the total cost of fabricating and assembling the product and to improve the chance to automate the process.
- Design verifiability into the product and its components to provide a natural test or inspection of the item.
- Avoid tight tolerances beyond the natural capability of the manufacturing processes and design in the middle of a part's tolerance range.
- Design "robustness" into products to compensate for uncertainty in the product's manufacturing, testing and use.
- Design for parts orientation and handling to minimize non-value-added manual effort, to avoid ambiguity in orienting and merging parts, and to facilitate automation.
- Design for ease of assembly by utilizing simple patterns of movement and minimizing fastening steps.
- Utilize common parts and materials to facilitate design activities, to minimize the amount of inventory in the system and to standardize handling and assembly operations.
- Design modular products to facilitate assembly with building block components and sub-assemblies.
- Design for ease of servicing the product.

2.3.3 DFM Guidelines for Castings

- On castings that must present a cosmetic appearance along with function, the exterior should be designed to follow simple, flowing lines with a minimum of projections.
- Avoid irregular or complicated parting lines whenever possible. Design for partings to be in one plane.
- Use ample but not excessive draft. Avoid any no-draft vertical surfaces unless there is no way out.
- Avoid long, slender cores through heavy metal sections or long spans. When unavoidable they should be straight and well anchored to the mold.
- Avoid the use of pattern letters on any surface other than the one parallel to the parting plane.

- Avoid sudden changes in section thickness that will create hot spots in the casting.
- Use ribs to stiffen or strengthen the castings, thus reducing weight and saving on material cost.
- Avoid sharp corners and fillet at junctions.

Figure 2.9 Examples of good and bad castings [Bralla, 1988]

- Stagger crossing ribs so that the junction will not create a hot spot, which could shrink.
- In case of die-casting, the cast surfaces that slide on the die cavity during die opening or ejection must have draft or taper to free the casting without binding.
- Also the inner walls of the die should have greater draft allowance as compared to the outer walls, as the former exert higher shrinkage forces.

Few examples showing good and bad designs for casting are depicted above in figure 2.9. These examples show the practical application of the above-mentioned guidelines.

2.4 Web-Based Applications for Castings

The application of web technology to the casting industry is gaining momentum. Some of these applications are discussed in this section.

Material and Process Selection:

Web-based material and process selection programs can eventually overcome the limitations of PC based software – such as the Cambridge Process Selector [Cebon, 2000] – by eliminating the cost and effort of installation, customizing, training, and maintenance and also it facilitates the implementation of the concurrent engineering philosophy irrespective of the geographical location of the team members. One example is the Manufacturing Advisory Service (MAS), which is useful for concept level manufacturing process and material selection [Smith, 2001]. It is available on line at: http://cybercut.berkeley.edu/mas2/.

The MAS is a *Java applet* (small application program) embedded in HTML pages, which performs its calculations using material-process capability data from a remote database. The user can enter the design specifications for one or more requirements and the weight for each requirement on a 1-5 scale [Figure 2.10]. The criteria for process selection includes batch size, dimensional tolerance, surface roughness, geometric shape, bounding box volume, wall thickness, material compatibility, production rate, setup time, setup cost and per part cost. Similarly, material selection criteria include cost per pound, yield strength, density, process compatibility, thermal expansion, elastic modulus and hardness. Each criterion is ranked on the scale of 0-100 based on its suitability, which is multiplied with its weight. The system provides feedback in terms of material possibilities, process possibilities and ranked material plus process suitability.

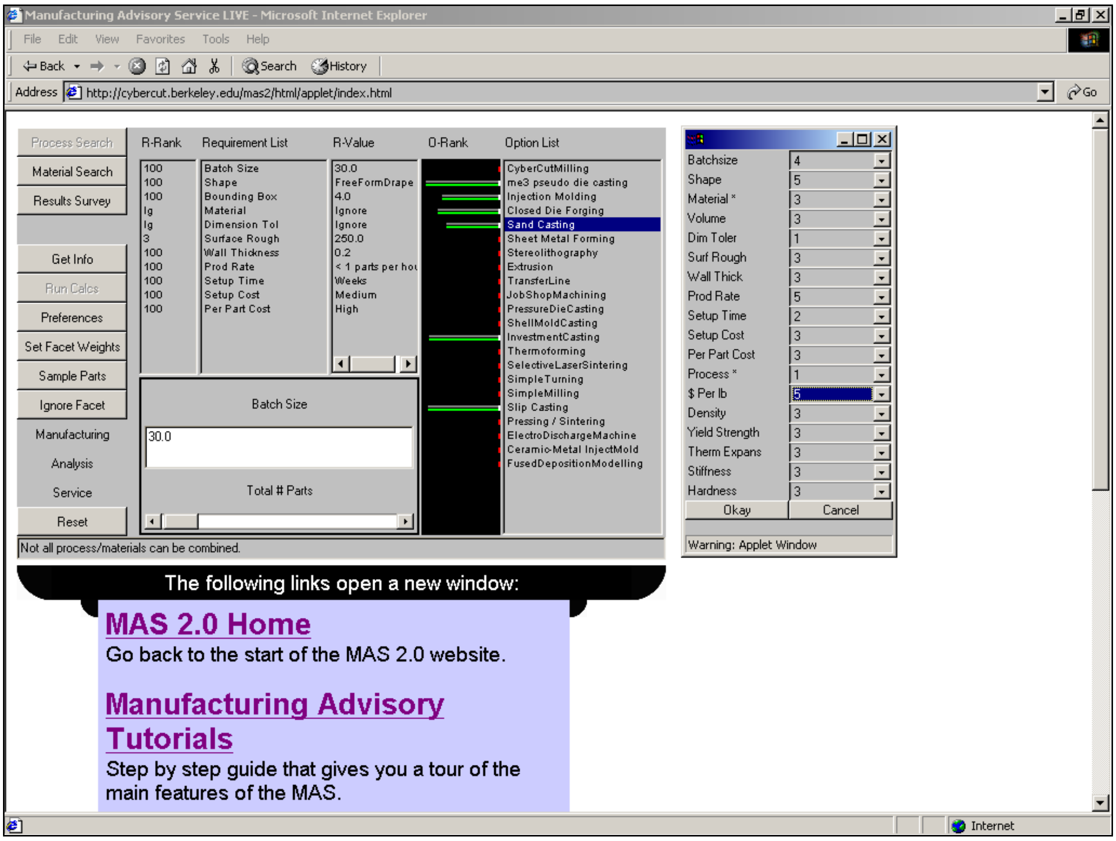


Figure 2.10 Material and process selection web page [Smith, 2001]

Internet-based Defect Analysis:

The defects analysis system is one of the few knowledge-based web-enabled applications immediately relevant to the casting industry. Rejections cost a foundry 3-20% of its revenue [Spada, 1998]. There are varieties of defects: surface (metal penetration, flash, sand fusion), discontinuity (misrun, cold crack, hot tear), dimensional (mismatch, warpage, distortion, core shift) and internal (shrinkage, gas porosity, blow holes). The expert system for defects analysis (available at http://207.250.195.21/4ebasic.htm) helps identify the cause of a particular defect, which otherwise may require an expert [Spada, 1988]. Its knowledge base, built with assistance by from industry experts worldwide, is stored in a decision tree. The system gathers the process and procedure information from the user and matches it against its knowledge base to identify the defect [Figure 2.11].

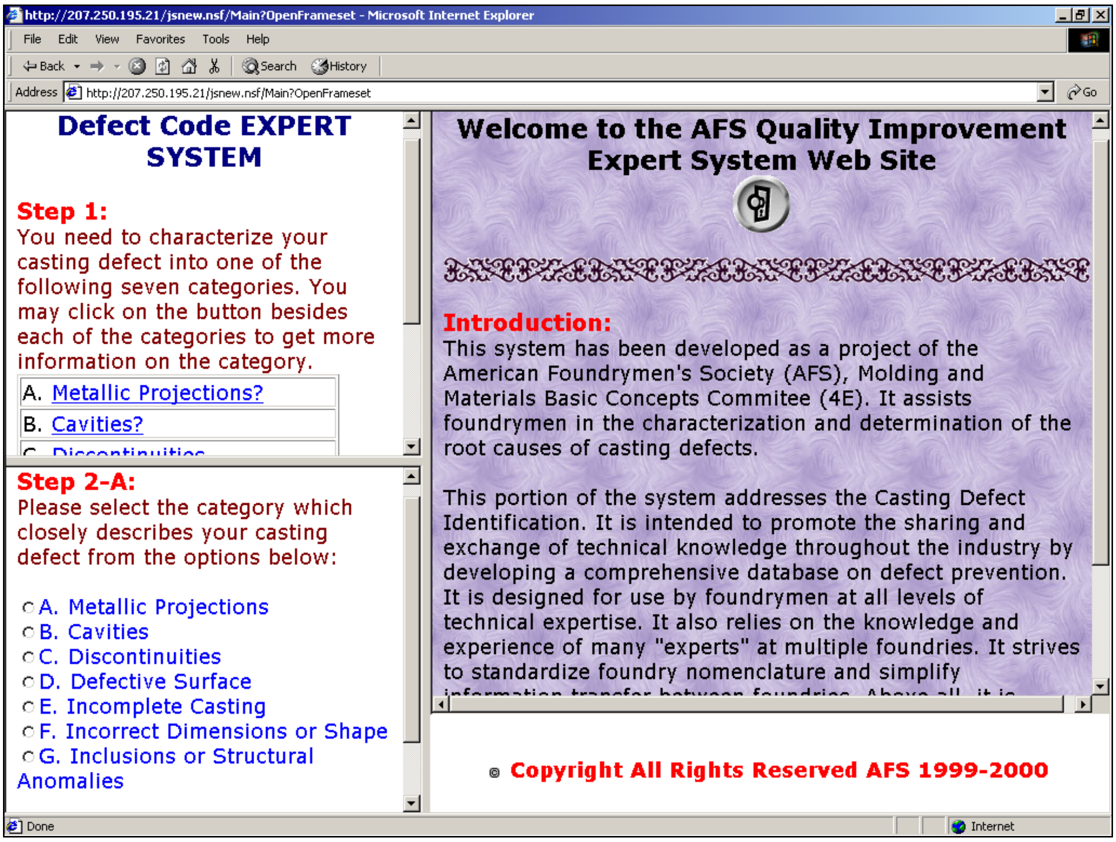


Figure 2.11 Casting defect analysis web page [AFS, 1999]

2.5 Summary

The manufacturing industry, over a period has seen evolution of different strategies. In 1970's, quality and TQM was the key to success. They were followed by Lean Manufacturing and Just-in-Time strategies in the 1980's. In today's highly competitive market of post WTO regime, the need of the hour is to develop quality products at low cost and in shorter lead times. Design-to-Cost is one tool, addressing this particular need of the industry. This methodology, which involves the setting up of a target cost for the product and then designing it within the set cost, requires tools, which could help give accurate cost estimates of the product in the early design stage. DFM guidelines act as a useful tool, in guiding the designers to design cost effective products. Further, with concurrent engineering concept gaining wide acceptance and enterprises setting up their facilities across the globe, in order to address the global market and avail the economies of scale, access to the required information to all the team members at the right time becomes very important. In such a global scenario, web-based collaborative engineering tools, which facilitate communication and sharing of information between the team members, would be the step in the right direction.

To implement the design-to-cost philosophy and to perform target costing an early cost estimation tool is required. Further, techniques like Activity Based Costing (ABC) requires a myriad of product related data which is not available at the early design stage. Therefore, need exists for an early cost estimation tool which would facilitate the implementation of the design-to-cost philosophy. Moreover, if such a tool were made available over the web, it would facilitate collaborative engineering. In the following chapter the overall system design has been discussed.

CHAPTER 3
SYSTEM DESIGN

This project has been developed on the backbone of the 'self-describing' XML compatible Casting Data Markup Language (CDML). The system that has been developed interacts with the database for storing and retrieval of data. Also the software developed during the course of this project, has been implemented as an application available on the WebICE framework. The CDML and WebICE framework are briefly described here, in the context of the cost estimation program development.

3.1 Casting Data Markup Language (CDML)

The Casting Data Markup Language (CDML) has been defined based on the standard XML [web-4.1]. It permits definition of domain specific tags and the data contained to become self-describing. The CDML is a template for storing life cycle data of the product. The approach to classify the casting project information in a hierarchical tree structure necessary for the CDML has been adopted from an earlier investigation [Akarte, 1996], which is then further improved and modified to develop the CDML to make it web compatible [12].

Information modeling for the web application has two principle requirements:

- Easy location of the required information and
- Fast retrieval over the internet.

Hierarchical structure enables easy retrieval of the required information. However, the speed of retrieval is dependent on the file size and access speed. The file size factor has been considered while defining the CDML structure.

The CDML consists of two parts: CDML tree and data blocks.

CDML tree: It represents the hierarchical relationship between different types of life cycle information involved in a casting project, whereas the *data blocks* are used for storing the actual project data. Important features of the CDML include systematic hierarchical classification of information for easy locating and viewing – tree for gross level and data blocks for detailed level information. It also supports hierarchical classification with a well-defined numbering scheme, which provides flexibility to incorporate additional information in tree as well the data blocks. Each type of information has been specified by the unique name and number that allow linking of tree with data blocks. Data block gives fast retrieval of information due to smaller size of 1 KB to 2 KB. Data block allows sharing of information. Separation of the tree (node) structure from the data blocks enables restructuring of the tree without modifying any associated programs using the data.

The CDML tree has been structured in the form of parent-child-grandchild for easy location of the information. Each node in the tree has a specific name and a number. For example, the top node is called PROJECT and it is numbered 001. Similarly, ADMIN node has a number 100. The unique numbering and naming approach enables: Linking of node with data block and, Flexibility to incorporate additional nodes in the tree. CDML data block is the compilation of detailed information pertaining to a particular type of node. Each data block is associated with a unique node of the CDML tree. In other words, there is one to one correspondence between the node of a CDML tree and a data block.

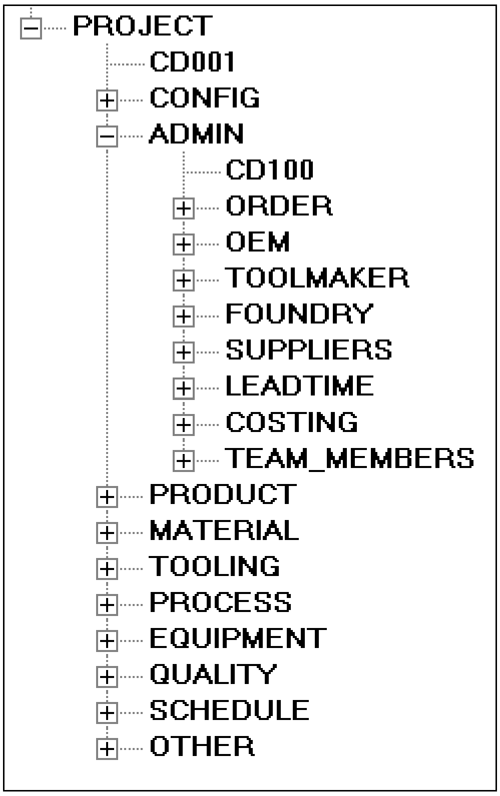


Figure 3.1 CDML tree [Akarte, 2002]

Each data block is stored in XML format, where the starting and ending tags are obtained from field names and the value residing between them is given by the field value. The use of XML for data storage also allows easy representation of units (Example, Kg/cm^3), if any, required for the specific casting information and they are attached to the starting tag. Following example shows two field names: pattern weight and shape complexity. The value of pattern weight is 2.5 while for shape complexity it is HIGH. The casting weight has a unit ‘Kg’ while shape complexity has no unit.

<PATTERN_WEIGHT UNIT="Kg"> 2.5 </PATTERN_WEIGHT >
<SHAPE_COMPLEXITY> HIGH </ SHAPE_COMPLEXITY >

Costing Block:

The costing block was defined based on the framework provided by CDML.

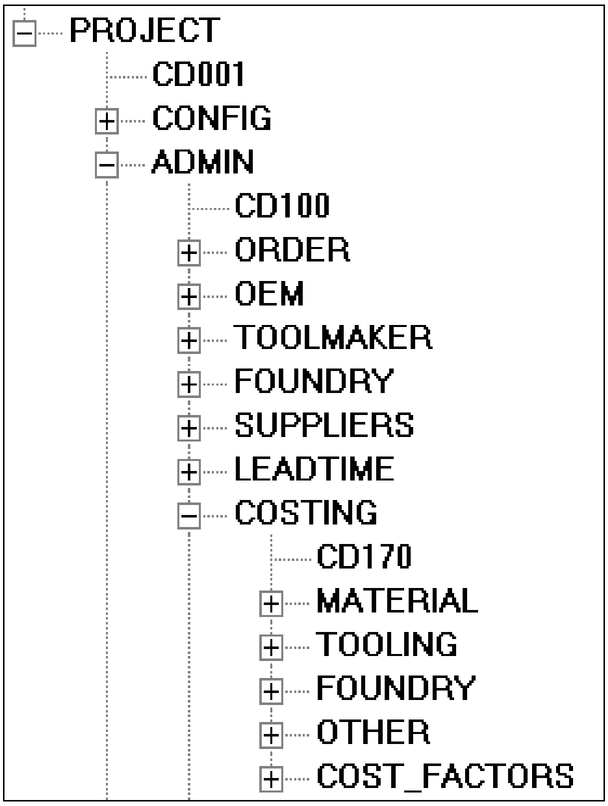


Figure 3.2 CDML tree (Cost Block)

This block was defined with the aim of supporting the cost estimation software in terms of storage and retrieval of data pertinent to the same. The different blocks within the cost block were identified based on the study of the various processes. The various blocks identified within the COST block are the MATERIAL, TOOLING, FOUNDRY, OTHER and finally COST FACTORS. The figure 3.3 shows the data block structure of the cost block. This is the first level of hierarchy of the cost block.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.1.0                              </VERSION>
<NAME>                 PROJECT.ADMIN.COSTING       </NAME>
<FILE>         CD170.XML                          </FILE>
<DATE>                 2001.05.05                         </DATE>
<TIME>         09:34:08                    </TIME>
<MODEL CREATED="NO">        NIL                    </MODEL>
<IMAGE CREATED="NO">        NIL                    </IMAGE>
<CAST_METAL UNIT="Rs">    46.33                    </CAST_METAL>
<TOOLING UNIT="Rs">       35.69                    </TOOLING>
<FOUNDRY UNIT="Rs">              57.26                    </FOUNDRY>
<OTHER UNIT="Rs">         42.83                    </OTHER>
<PREMIUM UNIT="Rs">               9.75                     </PREMIUM>
<GRAND_TOTAL UNIT="Rs">182.11                      </GRAND_TOTAL>
</CDML>
```


Figure 3.3 Sample data block ADMIN.COSTING


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
        <VERSION>              1.0                                 </VERSION>
        <NAME>                 ADMIN.COSTING.TOOLING       </NAME>
        <FILE>                 CD172.XML                   </FILE>
        <DATE>                        2001.05.05                   </DATE>
        <TIME>                 09:34:08             </TIME>
        <MODEL CREATED="NO">          NIL                  </MODEL>
        <IMAGE CREATED="NO">          NIL                  </IMAGE>
        <PATTERN UNIT="Rs">           4.18    </PATTERN>
        <CORE_BOX UNIT="Rs">                 0.00    </CORE_BOX>
        <DIE UNIT="Rs">                      0.00    </DIE>
        <INV_PATTERN UNIT="Rs">       0.00    </INV_PATTERN>
        <SAND_MOLD UNIT="Rs">         3.02    </SAND_MOLD>
        <SAND_CORE UNIT="Rs">         28.50   </SAND_CORE>
        <TOTAL_TOOLING UNIT="Rs">            35.69   </TOTAL_TOOLING>
</CDML>
```


Figure 3.4 Sample data block ADMIN.COSTING.TOOLING

Each block caters to a specific cost component of the casting. The MATERIAL block stores data regarding the cost of the raw metal and alloys. The FOUNDRY block has been further divided into three blocks viz., LABOUR, ENERGY and OVERHEADS. While the FOUNDRY block stores the cost data regarding the overall foundry cost in terms of labour, energy and overheads, the individual blocks within the FOUNDRY block store data corresponding to their field. The LABOUR cost block stores data regarding the labour cost data incurred during the different stages of the casting activity. Similarly the ENERGY block stores data regarding the energy cost incurred in producing the casting while the OVERHEAD costs data is stored in the OVERHEAD block. The OTHER block stores data regarding the delivery costs, taxes and the premium cost. The delivery costs include the packaging, transport, insurance and warehousing cost. The taxes include sales tax and customs. The figure 3.4 shows the tooling block, which is the second level in the cost block hierarchy. The figure 3.5 shows the third level of the cost block hierarchy.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>            1.0                                   </VERSION>
<NAME>               COSTING.FOUNDRY.ENERGY   </NAME>
<FILE>        CD174.XML                       </FILE>
<DATE>               2001.05.05                            </DATE>
<TIME>        09:34:08                  </TIME>
<MODEL CREATED="NO">     NIL                  </MODEL>
<IMAGE CREATED="NO">     NIL                  </IMAGE>
<BASIS>                                 PART    </BASIS>
<ELEC_POWER_RATE UNIT="Rs/kw-hr"> 5.0   </ELEC_POWER_RATE>
<MELTING UNIT="Rs">              24.75  </MELTING>
<OTHER_ENERGY UNIT="Rs">         8.66   </OTHER_ENERGY>
<TOTAL_ENERGY UNIT="Rs">         33.41  </TOTAL_ENERGY>
</CDML>
```


Figure 3.5 Sample data block COSTING.FOUNDRY.ENERGY

3.2 Web-Based Integrated Casting Engineering (WebICE)

WebICE (Web-based integrated casting engineering) has been designed to provide facilities for creating, viewing, modifying and updating casting project data (stored in CDML) over the Internet from any location worldwide. WebICE framework also supports application interface (by linking application program for decision making) as XML based data storage (CDML data block) enables easy extraction of the required casting life cycle data [12].

System Architecture: The architecture of webICE is shown in Figure 3.6. It consists of two parts: client side and server side. Client side has been used to provide the graphical user interface while databases and functions (programs) has been stored at server side. The cost estimation model developed is one of the functions that are stored on the server side.

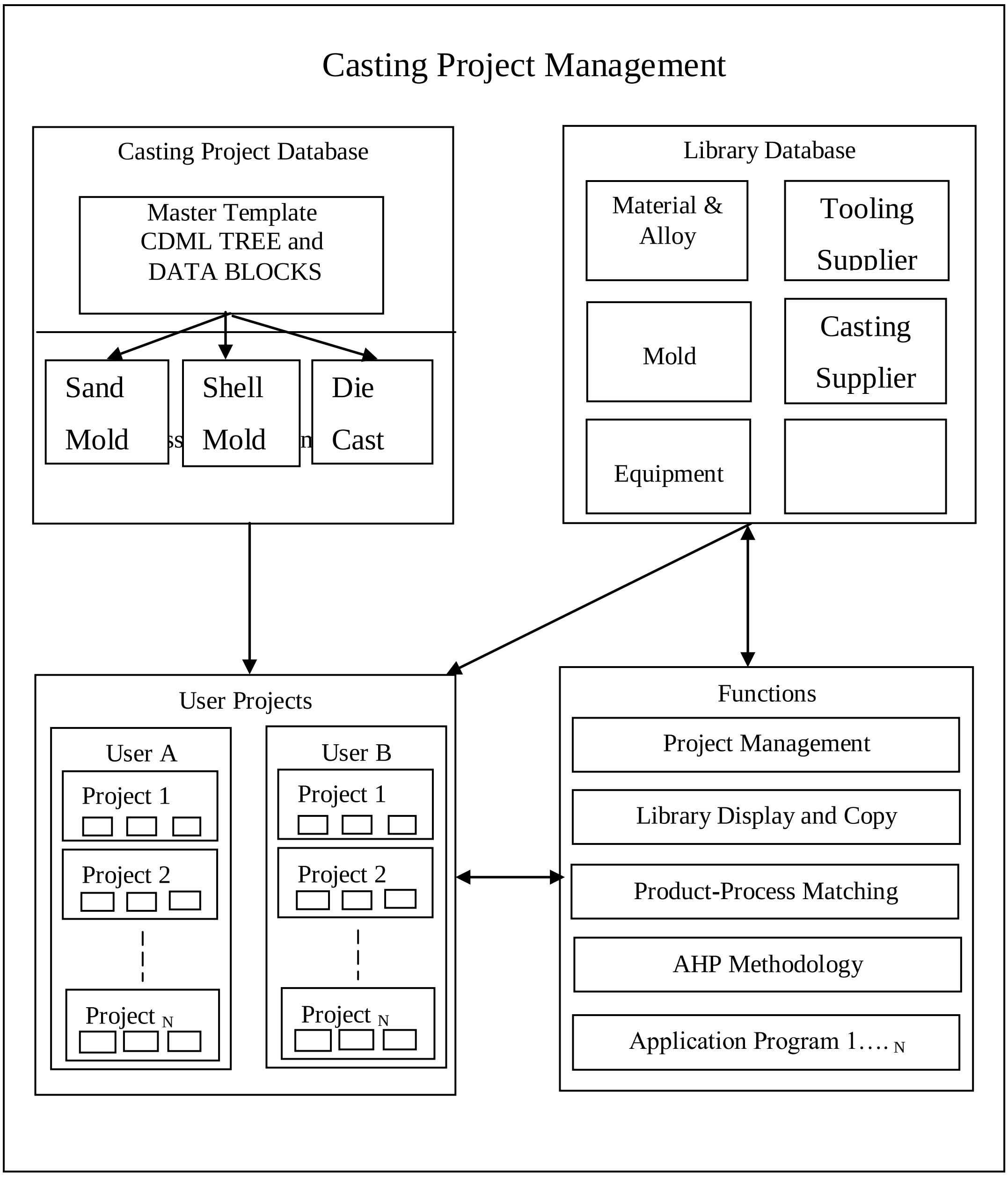


Figure 3.6 System architecture [Akarte, 2002]

Client-Server Approach:

Client-server computing is the logical extension of modular programming approach, where the program modules developed for the better interaction and maintenance can be placed and executed either completely at the server side or at the client side or distributed between them. The calling module is the client and it requires the service, whereas the called module is the server and it provides the service.

The client side consists of display of CDML tree, data blocks, functions, image/model, library options, user inputs and the results of the computations. The server side has four main components: casting project databases, library database, user projects and functions (application and supporting programs).

For better visual representation the screen display at client side is divided into seven windows and the output of various tasks performed by the webICE GUI are ported into different windows. The use of seven windows on the client-browser has been explained here. The first window is used to display the CDML tree, which allows the user to browse though it with the help of mouse clicks. On mouse click, the CDML tree will either expand or collapse. The second window is used to display the detailed information classified under a particular data block. It allows the user to browse and modify the required casting project data. The third window displays the name of the casting project. Fourth window shows the webICE GUI version. The fifth window, which is the largest, is used to display 2D images or 3D models. Other important use of this window is to display the application program and its interface; the computational results are also displayed in the same window. Casting project management functions (clone, delete, link and update) are ported in to the sixth window. The seventh window shows data block functions specific to a particular application, for example, Library and Calculate weights. Any application or library option attached to a specific data block will have a link through this window. The selection (by mouse click) of a particular function in this window will automatically initiate the respective program in window five.

Cost Function:

The cost function that has been developed during the course of this project is the application/function, which has been appended to the WebICE structure. The application has been so developed to be compatible with the WebICE structure. The cost program that has been developed, makes use of the XML based CDML structure, that has already been discussed in the previous section, as its backbone wherein the cost files are stored. The cost estimate and cost help functions which are linked to cost data block when clicked will execute the corresponding cost programs and the results of the estimate will be displayed in window five of the WebICE user interface. Also the results of the estimate will be saved in their corresponding cost component files when the save option is selected. Thus, the software making use of the WebICE framework facilitates the cost estimation of a casting, when all the pertinent inputs are specified in the right format within the cost files in the CDML structure.

3.3 Program Flow Diagram

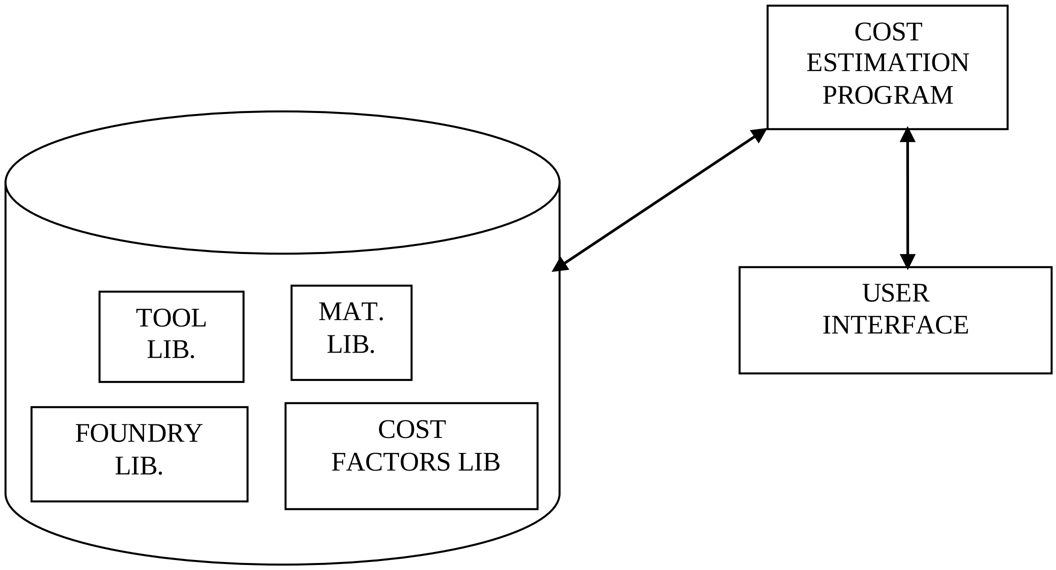


Figure 3.7 Program Flow Diagram

The cost estimation program interacts with the database and the cost factors library specific to that particular project/user and estimates the cost making use of the relevant input parameters and displays the output of the estimate in the window five of the WebICE user interface. The interface being user friendly allows the user to change the values of the input parameters as per his requirement and thereby analyze the change in the cost estimates of the product.

The program begins once the user provides all the required inputs and clicks the cost estimate button. The program reads all the necessary inputs from their respective files and makes use of this data to estimate the cost of the casting. Figure 3.8 shows the product file, which is one of the sources of input for the cost estimate program.

The data regarding alloy cost and the value of cost factors are read from their respective libraries. Once all the necessary inputs are provided the program estimates the cost based on the input data. The results of the estimate are displayed through the user interface of the WebICE framework. The results when saved are stored in their respective component files i.e., the foundry cost data gets stored in the foundry file, which can be viewed by expanding the foundry node within the cost block. Figure 3.9 shows the results of the estimate saved in its corresponding component file.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>               1.1.0                                   </VERSION>
<NAME>                  PROJECT.PRODUCT.REQUIREMENT             </NAME>
<FILE>                  CD230.XML                               </FILE>
<DATE>                  2001.05.05                              </DATE>
<TIME>                  09:34:08                                </TIME>
<MODEL CREATED="NO">        NIL                                 </MODEL>
<IMAGE CREATED="YES">       YOKE.jpg                            </IMAGE>
<PATTERN_WT UNIT="Kg">      2                       </PATTERN_WT>
<CORE_NUM>                  3                       </CORE_NUM>
<CORE_HOLE_SIZE_MIN UNIT="mm">19.2                  </CORE_HOLE_SIZE_MIN>
<TOTAL_CORE_WT UNIT="Kg">         3.0               </TOTAL_CORE_WT>
<SHAPE_COMPLEXITY>                MEDIUM            </SHAPE_COMPLEXITY>
<NUMBER_OF_CAVITIES>              3                 </NUMBER_OF_CAVITIES>
<ROUGHNESS_MAX UNIT="um">         6.0               </ROUGHNESS_MAX>
<TOLERANCE_REQD UNIT="mm">        0.4               </TOLERANCE_REQD>
<POROSITY_MAX UNIT="mm">          1.0               </POROSITY_MAX>
<TREATMENT_REQD>                  NO                </TREATMENT_REQD>
<MACHINING_REQD>                  NO                </MACHINING_REQD>
<PRODUCTION_RATE UNIT="No/Hour">55                  </PRODUCTION_RATE>
<ORDER_SIZE UNIT="Qty/year">      12000             </ORDER_SIZE>
<DEL_QUANTITY UNIT="Qty/Order">   1000              </DEL_QUANTITY>
<DEL_FREQUENCY UNIT="Orders/Month">1                </DEL_FREQUENCY>
<DATE_LOI>                        0                 </DATE_LOI>
<DATE_QUOTE>                      0                 </DATE_QUOTE>
<DATE_SAMPLE>                     0                 </DATE_SAMPLE>
<PRIORITY>                        HIGH              </PRIORITY>
<FIXED_COST UNIT="Rs">            150000            </FIXED_COST>
<STORAGE_DAYS>                    NIL               </STORAGE_DAYS>
<DISTANCE UNIT="KM">              NIL               </DISTANCE>
<SAND_CORE_MATERIAL>        OIL_SAND                </SAND_CORE_MATERIAL>
<INSPECTION>                      YES               </INSPECTION>
<JIT_DELIVERY>                    YES               </JIT_DELIVERY>
<ESTIMATED_COST UNIT="Rs/part">0                    </ESTIMATED_COST>
</CDML>
```


Figure 3.8 Product requirement file

```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
        <VERSION>      1.0                                   </VERSION>
        <NAME>         ADMIN.COSTING.OTHER                   </NAME>
        <FILE>         CD177.XML                             </FILE>
        <DATE>         2001.05.05                            </DATE>
        <TIME>         09:34:08                              </TIME>
        <MODEL CREATED="NO">      NIL                        </MODEL>
        <IMAGE CREATED="NO">      NIL                        </IMAGE>
        <BASIS>                    BATCH          </BASIS>
<PACKAGING UNIT="Rs">              4.18           </PACKAGING>
<TRANSPORT UNIT="Rs">              0.00           </TRANSPORT>
<INSURANCE UNIT="Rs">              1.04           </INSURANCE>
<WAREHOUSE UNIT="Rs">              0.00           </WAREHOUSE>
<TOTAL_DELIVERY UNIT="Rs">         5.22           </TOTAL_DELIVERY>
<SALES_TAX UNIT="Rs">              5.57           </SALES_TAX>
<CUSTOM UNIT="Rs">                 22.29          </CUSTOM>
<TOTAL_TAXES UNIT="Rs">            27.86          </TOTAL_TAXES>
<PREMIUM UNIT="Rs">                9.75           </PREMIUM>
<TOTAL_OTHER UNIT="Rs">            42.83          </TOTAL_OTHER>
</CDML>
```


Figure 3.9 OTHER_COST file

Thus, the data blocks pertaining to cost estimation have been defined within the framework of the Casting Data Markup Language (CDML). Further, the cost estimation program will be made compatible with the WebICE framework, which facilitates collaborative engineering over the web. In the next chapter, the cost estimation model has been explained and the results of the regression analysis for tooling cost have been discussed.

CHAPTER 4

CASTING COST ESTIMATION MODEL

The cost model that has been developed in this work has been discussed in this chapter. In the last section of this chapter the results of the regression analysis for tooling cost, which happens to be the most important cost component of the product, have been discussed. The total cost has been broken down into different cost components, which are calculated individually first and added to give the total cost. The various cost components within the cost model have been discussed in detail in the following sections.

4.1 Cast Metal Cost

The cast metal cost is made up of the metal cost and the alloy cost.

$$Cast\_Metal\_cost = Metal\_cost + Alloy\_cost \tag{1.0}$$

a) Metal cost:

The metal cost is again made up of raw material cost and scrap cost.

$$Metal\_cost = Raw\_material\_cost + Scrap\_cost \tag{1.1}$$

The raw material and scrap costs are calculated based on their percentage contribution to the total casting weight, the wastage rate and their corresponding rates.

$$Raw\_material\_cost = 0.01*(\%raw\_material + \%wastage)*Casting\_wt*metal\_rate \tag{1.2}$$

$$Scrap\_cost = 0.01*(\%\ scrap + \%\ wastage)*Casting\_wt*Scrap\_rate \tag{1.3}$$

The factors such as the wastage and the percentage of raw material and scrap will depend on the practice that is followed in a particular foundry.

b) Alloy cost:

The alloy cost is calculated from the equivalent alloy weight and the equivalent alloy cost. This is obtained by the weighted average of the alloy weights and their respective costs.

If say w1, w2 and w3 are the weights of three alloys and their corresponding costs are c1, c2, and c3 then the equivalent alloy weight and equivalent alloy costs are given by

$$Equ\_alloy\_wt=(w1*c1+w2*c2+w3*c3)/(c1+c2+c3) \quad (1.4)$$

$$Equ\_alloy\_cost==(w1*c1+w2*c2+w3*c3)/(w1+w2+w3) \quad (1.5)$$

And the alloy cost will be given by

$$Alloy\_cost= Equ\_alloy\_wt* Equ\_alloy\_cost \quad (1.6)$$

4.2 Tooling Cost

The tooling cost is made up of the permanent tooling cost and the dispensable tooling cost, which are supplementary to each other.

$$Tooling\_cost=Permanent\_tooling\_cost+Dispensable\_tooling\_cost \quad (2.0)$$

a) Permanent Tooling cost:

The permanent tooling cost is made up of the pattern cost, the core cost and the die cost.

$$Permanent\_tooling\_cost=Pattern\_cost+Core\_cost+Die\_cost \quad (2.1)$$

The pattern cost is calculated using the pattern weight, pattern material cost, pattern manufacturing cost, number of patterns and the order size.

$$Pattern\_cost=((Pattern\_wt* Mat\_rate + Pattern\_manuf\_cost) * no\_of\_pattern) / Order\_quantity \quad (2.2)$$

$$Core\_cost=Core\_box\_wt * Mat\_rate + Core\_box\_manuf\_cost \quad (2.3)$$

$$Die\_cost=Die\_insert\_cost + Die\_block\_cost + Die\_manuf\_cost \quad (2.4)$$

The pattern and core box manufacturing cost depends on the material, complexity and accuracy required. Also, the insert and the core contribute the major cost of a die. The die cost will depend on the no of cores, the complexity of the cores and their direction of movement.

b) Dispensable Tooling cost:

The dispensable tooling cost is made up of investment pattern cost, sand core cost and sand mold cost.

Dispensable_tooling_cost=Inv_pattern_cost+Sand_core_cost+ Sand_mold_cost (2.5)

The investment pattern cost is given by

*Inv_pattern_cost=Pattern_wt * Mat_rate + Patern_manuf_cost* (2.6)

*Core_cost=(Core_wt*Mat_rate+Core_manuf_cost)* No_of_cores* (2.7)

*Mold_cost=Mold_wt*Mat_rate+Mold_manuf_cost* (2.8)

Here again, the pattern, core and mold manufacturing cost will depend on the complexity and the accuracy required.

4.3 Foundry Cost

The foundry cost is made up of the energy cost, the labor cost and the fixed costs.

Foundry_cost=Energy_cost+Labor_cost+Overhead_cost (3.0)

a) Energy cost:

The energy cost is obtained as a product of energy rate and total energy consumed. The total energy is the sum of the melt energy and other energy, (which is taken as a percentage of melt energy).

*Energy_cost=(Melt_energy+Other_energy)*Energy_rate* (3.1)

The other energy is made up of the molding cost, holding cost, heat-treatment cost etc.

b) Labor cost:

The labor cost is calculated based on three stages of activities namely, pre-processing, casting and post-processing. Cost for each stage is calculated on the basis of the time taken for that stage, the wage rate and the batch size.

Labor_cost=Pre-processing_cost+casting_cost+Post_processing_cost (3.2)

*Pre_processing_cost=(Time_taken*Wage_rate)/Batch_size* (3.3)

*Casting_cost=(Time_taken*Wage_rate)/Batch_size* (3.4)

Post_processing_cost=(Time_taken*Wage_rate)/Batch_size (3.5)

*other_cost=(Time_taken*Wage_rate)/Batch_size* (3.6)

The time taken in each case will depend on the activities involved in each stage and the task times of these activities.

c) Overhead cost:

The fixed cost is made up of the depreciation cost interest cost and the other cost. All these costs are taken on per kg basis.

Overhead_cost=Depriciation_cost+Interest_cost+Fixed_cost (3.6)

*Depriciation_cost=Depriciation_cost_rate*Casting_wt* (3.7)

*Interest_cost= Interest_cost_rate*Casting_wt* (3.8)

*Fixed_cost= Fixed_cost_rate * Casting_wt* (3.9)

The depreciation cost depends on the method used to calculate depreciation, the equipment cost and its life. The rate of interest and the amount of loan decide the interest cost rate. These equations have been used to estimate the cost of the individual cost elements and thereby the total cost of the casting.

4.4 Other Cost

The other cost is made up of delivery cost and the tax cost.

Other_Cost=Delivery_cost+Taxes+Premium_cost (4.0)

a) Delivery Cost:

The delivery cost is made up the packaging cost and transportation cost.

Delivery_cost=Packaging_cost+Transport_cost (4.1)

The packaging cost is taken as a percentage of the part cost.

*Packaging_cost=0.01*packaging_rate*(material_cost+tooling_cost+foundry_cost)* (4.2)

*Transport_cost=0.01*transport_rate*(distance*part_weight*order_size)* (4.3)

Warehousing_cost=warehousing_rate(storage_days*part_weight*order_size)* (4.4)

b) Taxes:

The taxes include the sales tax, customs and the warehousing cost of the final product.

Taxes_cost=Sales_tax_cost+Custom_cost+Warehousing_cost (4.5)

$$Sales\_tax\_cost = 0.01 * sales\_tax\_rate * (material\_cost + tooling\_cost + foundry\_cost) \quad (4.6)$$

$$Custom\_cost = 0.01 * custom\_rate * (material\_cost + tooling\_cost + foundry\_cost) \quad (4.7)$$

c) *Premium Cost:*

The premium cost is calculated by considering additional cost to the total factory cost if either one or all of priority, just-in-time and inspection are required.

$$Premium\_cost = 0.01 * (priority\_rate + just\text{-}in\text{-}time\_rate + inspection\_rate) * (material\_cost + tooling\_cost + foundry\_cost) \quad (4.8)$$

Finally the total cost of the product is given by the following equation.

$$Total\_Cost = Cast\_Metal\_cost + Tooling\_cost + Foundry\_cost + Other\_Cost \quad (5.0)$$

These equations have been used to estimate the cost of the individual cost elements and thereby the total cost of the casting.

4.5 Regression Analysis for Tooling Cost

Of all the cost components, the tooling cost is the most difficult to estimate especially when doing an early cost estimation when very little information is available. Therefore, a regression analysis was undertaken to determine a relationship for estimating the tooling cost. Regression analysis uses product parameters as variables in cost functions. This form of estimation requires substantial amount of historical data rather than personal experience. Starting with production records of many parts, the estimator searches for correlations between the cost of the final part and one or more parameters of the part. A graph of a likely parameter viz., weight, volume etc is plotted against the cost of the part. If the regression of the data has a sufficiently high correlation coefficient, then the curve fit can be used to make future cost predictions given the value of the chosen parameters.

Selecting the Parameters:

In order to perform the regression analysis and derive a generic relationship between the cost of the tooling and the affecting parameters, it is first required to identify the factors that affect the cost of the tooling. Based on the study of the various processes, three parameters were identified, which had a substantial influence on the cost of the tooling. These three parameters are, the weight of the pattern/die, the complexity of the tooling and finally the accuracy required. Having identified the significant parameters, the next step was to collect actual historical data from the industry.

Data Collection:

Large amount of historical data is required to perform a regression analysis, so that the results of the analysis are consistent and generic in nature and can be applied over a wide range of variation of the affecting parameters.

Table 4.1 Wooden (Teakwood) pattern data

SR NO.	PART	WT (Kg)	COMP IND	ACC IND	COST (Rs)
1	NOZZLE	31	40	60	9500
2	TOOTH	35	60	80	13500
3	VALVE BODY	47	40	40	23500
4	CONICAL LINING PLATE	76	80	60	3900
5	16" BUTTERFLY VALVE	99	20	20	24500
6	GATE VALVE BODY 8"	129	40	40	16500
7	GEAR BOX	300	60	40	15000
8	SEGMENT FEEDER	325	60	80	56000
9	GEAR WHEEL	400	20	40	8000
10	DRUM GEAR	500	40	20	10000
11	GEAR WHEEL	500	40	20	10000
12	BALARD FOR PORT	500	40	60	25000
13	BOWL	625	20	40	43000
14	SOCKET RING	800	40	60	30000
15	BOWL	800	40	60	50000
16	ROPE DRUM	1000	40	20	20000
17	LOWER PLATE	1000	40	20	75000
18	RING	1000	40	60	100000
19	WIND MILL HUB	2225	40	60	260000

The data that was required to be collected involved the weight of the tooling, its complexity, and the accuracy required in the manufacture of the tooling and most significantly the cost of the tooling as incurred by the toolmaker. The complexity and accuracy data was converted onto a scale of 0-100 with 20 being simple and 80 being very complex or very accurate as may be the case. Data collected both for wooden and aluminium patterns have been listed below in table 4.1 and 4.2 respectively. The above table gives a list of 19 wooden patterns that have been manufactured in the industry. It also gives details of the pattern in terms of its weight in kilograms, the complexity index on a scale of 0-100 (0 being least complex and 100 being most complex), accuracy also on a scale of 0-100 (0 being normal and 100 being very accurate) and the cost in Rupees as incurred by the manufacturer. The pattern material is teakwood, which is the most common wood used for making wooden patterns.

The following table gives a list of 12 metal patterns, which have been manufactured in the industry, and the details of the pattern in terms of its weight in kilograms, the complexity index and accuracy index on a scale of 0-100, similar to the previous table and finally the cost in Rupees as incurred by the manufacturer. The pattern material in this case is aluminium, as it is the most common material used for metal patterns, because of its high strength to weight ratio.

Table 4.2 Metal (Aluminium) pattern data

SR NO.	PART NAME	WT KG	COMP IND	ACC IND	COST (Rs)
1	COVER	2	20	20	2950
2	300MM BUTTERFLY	17	40	40	9180
3	125 VALVE BODY	19	40	40	10600
4	HOUSING	24	60	40	28000
5	COVER	60	80	40	32000
6	HAMMER	71	40	20	18200
7	COVER PLATE	100	20	60	10000
8	HAMMER	103	40	20	21450
9	BALL VALVE 8"	115	60	40	32000
10	CHAIN LINK	115	60	60	30000
11	PUMP CASING	500	60	60	50000
12	HEAD	570	80	80	258000

Regression Analysis for Wooden Pattern:

The regression analysis for wooden pattern was conducted using the data that has been collected from industry sources. First of all a correlation between the cost and the identified parameters was conducted, the results of which are shown in the table 4.3.

Table 4.3 Correlation matrix for wooden pattern

	WEIGHT	COMPLEXITY INDEX	ACCURACY INDEX	COST
WEIGHT	1			
COMPLEXITY INDEX	-0.1993	1		
ACCURACY INDEX	-0.0238	0.478354	1	
COST	0.8672	-0.08584	0.215097	1

A study of the results of the correlation matrix shows that weight has a very high correlation coefficient with the cost of the product, which implies that a major part of the total cost is explained by the material cost of the product. Another noteworthy observation of the results is that, for the given set of data complexity index has a negative correlation coefficient with cost, which implies that a regression analysis with given data will give a negative coefficient for complexity, which shall be verified by the results of the regression analysis. This implies that for the given set of data as the complexity increases the cost of the product decreases, which is in direct contradiction of what is expected. The same will be reflected in the results of the regression analysis. First, a regression analysis was conducted making use of all the three parameters namely, weight, complexity index and accuracy index. A linear relationship has been assumed between the parameters and the cost of the product. The results of the same have been depicted in the table 4.4 and 4.5.

Table 4.4 Summary output for regression analysis of wooden pattern

Regression Statistics	
Multiple R	0.87958127
R Square	0.77366321
Adjusted R Square	0.682871112
Standard Error	29544.75475
Observations	19

From the above table we see that R square has a value of 0.77 (approx), which implies that 77% of the cost variations are explained by the cost model. The following table gives the values of the coefficient of the various parameters based on the given set of data. It also gives p-values for the various parameters that have been considered in the analysis. Since the regression analysis is being conducted for a cost analysis equation, the intercept has been taken as zero in the model.

Table 4.5 Regression analysis coefficients for wooden pattern

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	86.33142151	12.08609919	7.143034	2E-06
COMPLEXITY INDEX	-687.227721	422.6356463	-1.62605	0.123
ACCURACY INDEX	569.8036636	382.5399037	1.489527	0.156

From the table we see that, complexity coefficient has a negative value which is consistent with the predictions made on the basis of the results of correlation that was conducted earlier. As per the regression analysis fundamentals, only those parameters, for which the p-value is either less than or close to 0.05, are considered significant. As per this rule, we observe that in the present model only weight turns out to be a significant parameter while the others are not significant. This again is contradictory to the established fact that complexity and accuracy affect the cost of pattern significantly. This variation of the results could be due to the small size of the sample data.

Cost = 86.33 * Weight - 687.23 * Complexity_index +569.8 *Accuracy_index

A larger set of data could possibly have given a more consistent and generic result, in accordance with the established facts. In order to ascertain the impact of accuracy, a regression analysis was performed considering only weight and complexity as the independent parameters and cost as the dependent parameter. The results of the same are shown in table 4.6 and 4.7

Table 4.6 Regression analysis considering only weight and complexity

Regression Statistics	
Multiple R	0.861555272
R Square	0.742277486
Adjusted R Square	0.668293809
Standard Error	30585.42528
Observations	19

A study of the above table shows that R square has a value 0.74 (approx) which implies that 74% of the variations in cost are explained by the model. This is just marginally less than the previous model, which reinstates the results that accuracy is not a significant parameter, because its omission has not affected the results significantly.

Table 4.7 Regression coefficients considering only weight and complexity

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	90.44536364	12.18074293	7.425275	1E-06
COMPLEXITY INDEX	-131.913556	206.0897397	-0.64008	0.531

The table 4.7 shows the values of the coefficients for weight and complexity coefficient. The coefficient for complexity index is again negative as expected and its p-value is high which again indicates that it is not a significant parameter either. The equation for cost based on the above results is given below.

Cost = 90.45* Weight - 131.91* Complexity_index

In order to generate a model consistent with the established facts a regression analysis was performed omitting the complexity index as it gave a negative coefficient. The results of this regression analysis are shown in table 4.8 and 4.9.

Table 4.8 Regression analysis considering only weight and accuracy

Regression Statistics	
Multiple R	0.858056174
R Square	0.736260398
Adjusted R Square	0.661922774
Standard Error	30940.40665
Observations	19

From the table we see that R square has value of 0.736 (approx), which implies that 73.6% of the cost variations are explained by cost model. This is almost same as that of the previous models, which again indicates that accuracy is not a significant parameter.

Table 4.9 Regression coefficients considering only weight and accuracy

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	84.49270525	12.60150999	6.704967	4E-06
ACCURACY INDEX	21.10307707	188.7028611	0.111832	0.912

From the above table we see that though accuracy index has a positive value of the regression coefficient, its p-value is very high which implies that it has very little significance. The resulting equation is shown below.

Cost = 84.49 * Weight + 21.1 * Accuracy_index

Further analysis was also conducted for model with accuracy having a relation, power 1.5 and 1.9, the results of which have been shown in the following tables.

Table 4.10 Regression analysis considering only weight and accuracy ($A^{1.5}$)

Regression Statistics	
Multiple R	0.859564445
R Square	0.738851034
Adjusted R Square	0.664665801
Standard Error	30788.0724
Observations	19

From the above table, we see that for accuracy to the power 1.5 the R square value has improved over the previous model, though not significantly.

Table 4.11 Regression coefficients considering only weight and accuracy ($A^{1.5}$)

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	82.30318681	11.91269999	6.908861	3E-06
ACCURACY_INDEX1	9.778090749	22.96611004	0.425762	0.676

From this table we see that the coefficients are positive and the p-value of accuracy index has improved over the previous model though not significantly.

$$\textbf{Cost} = 82.3 * \textbf{Weight} + 9.78 * \textbf{Accuracy-index}^{1.5}$$

Table 4.12 Regression analysis considering only weight and accuracy ($A^{1.9}$)

Regression Statistics	
Multiple R	0.861032631
R Square	0.741377191
Adjusted R Square	0.667340555
Standard Error	30638.8003
Observations	19

The table 4.12 shows the results of the regression analysis for accuracy to the power 1.9. Here again the R square value has improved over the previous models though not significantly. This is because for the given set of data accuracy index has little effect on the cost and so even by increasing its power from 1.5 to 1.9 the R square value has not improved much.

Table 4.13 Regression coefficients considering only weight and accuracy ($A^{1.9}$)

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	81.51220422	11.45098525	7.118357	2E-06
ACCURACY_INDEX2	2.47146803	4.182954673	0.590843	0.562

From table 4.13 we see that the p-value has further improved for accuracy index though very little. The coefficients however like in previous cases are positive. The equation resulting from the above analysis is given below.

$$\textbf{Cost = 81.51 * Weight + 2.47 * Accuracy\_index}^{1.9}$$

Regression Analysis for Metal Pattern:

The regression analysis for metal pattern was conducted making using of the data collected from industry sources. First of all a correlation was conducted between cost and the independent parameters. The results of same are shown in table 4.14

Table 4.14 Correlation matrix for metal pattern

	WEIGHT	ACCURACY INDEX	COMPLEXITY INDEX	COST
WEIGHT	1			
ACCURACY INDEX	0.7172	1		
COMPLEXITY INDEX	0.5123	0.484881	1	
COST	0.8032	0.671025	0.59782	1

From the above table we see that weight has a high value of coefficient with cost, as was the case for wooden pattern. In addition, in this case the complexity coefficient has a positive value of the coefficient, which is consistent with the established facts.

Table 4.15 Summary Output for Regression Analysis of Metal Pattern

Regression Statistics	
Multiple R	0.8103
R Square	0.6565
Adjusted R Square	0.4691
Standard Error	44916
Observations	12

A regression analysis assuming a linear relationship of the parameters with the cost was conducted the results of which are shown in table 4.15 and 4.16.

From the table we see that R square has a value 0.656 (approx), which implies that 65.6% of the variations in cost are explained by the model. This however is less than that obtained for wooden pattern. The following table gives the coefficients of the parameters. We see that for the assumed linear relationship of the parameters, the accuracy coefficient has a negative value. Also the p-values for complexity and accuracy are very high which implies that these two parameters are not significant.

Table 4.16 Regression analysis coefficients for metal pattern

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	274.955573	91.35330972	3.0098	0.0147205
ACCURACY INDEX	-179.71714	839.3734225	-0.21411	0.8352345
COMPLEXITY INDEX	293.633508	655.5861787	0.44789	0.6648162

The resulting equation of the analysis is given below.

Cost=274.96*Weight-179.7*Accuracy_index+293.63*Complexity_index

One more analysis was conducted with a relationship having accuracy to the power 1.5. The results of the same are shown in the following tables. From table 4.17 we see that R square has a value 0.658 (approx) which implies that 65.8% of the cost variations are explained by the cost model, which indicates a very insignificant improvement over the previous model.

Table 4.17 Regression analysis for metal pattern ($A^{1.5}$)

Regression Statistics	
Multiple R	0.81152405
R Square	0.65857129
Adjusted R Square	0.47158713
Standard Error	44781.0794
Observations	12

From table 4.18 we see that now all the coefficients are positive which is consistent with the established facts. Also the p-value for accuracy index has improved over the previous model while the p-value for complexity index has worsened.

Table 4.18 Regression coefficients for metal pattern ($A^{1.5}$)

	Coefficients	Standard Error	t Stat	P-value
INTERCEPT	0	#N/A	#N/A	#N/A
WEIGHT	245.844452	103.2416107	2.38125	0.0411431
ACCURACY_INDEX1	32.0606379	101.2531173	0.31664	0.758739
COMPLEXITY INDEX	48.776925	528.4058329	0.09231	0.9284742

The equation resulting from the above analysis is given below.

$$\textbf{Cost} = 245.8 * \textbf{Weight} + 32.06 * \textbf{Accuracy\_index}^{1.5} + 48.78 * \textbf{Complexity\_index}$$

From the analysis of the above results, it is obvious that the results obtained in case of wooden pattern were not consistent with the established facts as the complexity coefficient had a negative value. However, this analysis shows that the material cost in case of metal pattern will be more than that of a wooden pattern for the same weight, because the coefficient for weight is greater in case of metal pattern. This observation is consistent with the established facts. The results of the above analysis would have been more useful if more set of data was available for performing the regression analysis. However, the above study is helpful in understanding the effect of the chosen parameters on the tooling cost.

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Database and Cost Factors Library

The database for cost is designed on similar lines to that of the entire CDML database. The tree structure with data blocks is same as that discussed previously in the section on CDML. The COST block comes under the ADMIN block within the PROJECT BLOCK. Within each data file a three level hierarchy is maintained i.e. the child, the parent and the grandparent names are stored in the nametag of the file. Within the COST block are again the MATERIAL, TOOLING, FOUNDRY and OTHER blocks. Each block stores the results related to that particular block. However, the first eight tags remain the same for each file as described earlier. Again, within the FOUNDRY block there are three blocks viz. ENERGY, LABOUR and OVERHEADS, each storing the results corresponding to their own block. The tree structure of the costing database is shown in the figure 3.2 [Akarte, 2002].

```
<PROJECT>
<ADMIN>......
        ................

        <LEADTIME><CD160/>
                .........
                <DELIVERY>      <CD164/>        </DELIVERY>
        </LEADTIME>
        <COSTING><CD170/>
                <MATERIAL>      <CD171/>        </MATERIAL>
                <TOOLING>       <CD172/>        </TOOLING>
                <FOUNDRY>       <CD173/>
                                <ENERGY>        <CD174/>        </ENERGY>
                                <LABOUR>        <CD175/>        </LABOUR>
                                <OVERHEADS>     <CD176/>        </OVERHEADS>
                </FOUNDRY>
                <OTHER> <CD177/>
                </OTHER>
                <COST_FACTORS>          <CD179/>        </ COST_FACTORS >

        </COSTING>
        .............
        .....
</ADMIN>
</PROJECT>
```

Figure 5.1 CDML tree file format (Cost Block)

The cost factors library stores the default values of the various cost factors, which are used for the process of cost estimation. These factors are used along with the user inputs to derive the cost estimate. The default values of these factors eliminate the need for their entry by the user manually. The typical cost factors are the packaging rate, the transportation rate, the tax rate, warehousing rate etc.

The value of these factors remains constant over the range of products and therefore these values are stored in a different file called the cost factors file. However, like the other files, the values of the factors in this file can also be modified, if the user wishes to do so as per his requirement.

5.2 Implementation Tools

A variety of application tools has been used for developing the software model and web-enabling it. The client side interface has been developed using HTML, JavaScript and XML-DOM functions while on the server side PHP (platform independent scripting language) has been used.

HTML (Hyper Text Markup Language) is a scripting language used to develop static web pages. It has a set of pre-defined tags, which are used for the purpose of data representation. The user interface is designed used this application tool.

JavaScript is again a scripting language, but it allows dynamic changes in the web page. It supports various dynamic functions like OnClick etc. It thus facilitates dynamic design of the web pages.

XML (eXtensible Markup Language) is used for the database. It is used as a meta language to define language. Due to its flexibility that allows the user to define tags of his choice, it is very useful. It is actually an extension of HTML.

PHP is again a scripting language, for the server side. It interacts with the database and various functions on the server side. Its syntax being very similar to programming language C, it is very easy to use.

5.3 User Interface

The user interface has been designed using HTML (Hyper Text Markup Language). The user interface has been divided into seven windows. The display of each of the windows has already been discussed in the earlier chapter. The fifth window, which is the largest, displays the computational results of the cost estimation. This particular window has two different interfaces. One showing the summary of the cost estimation and the second showing the detailed results of the cost analysis. The figure 5.2 shows a screen print of the user interface.

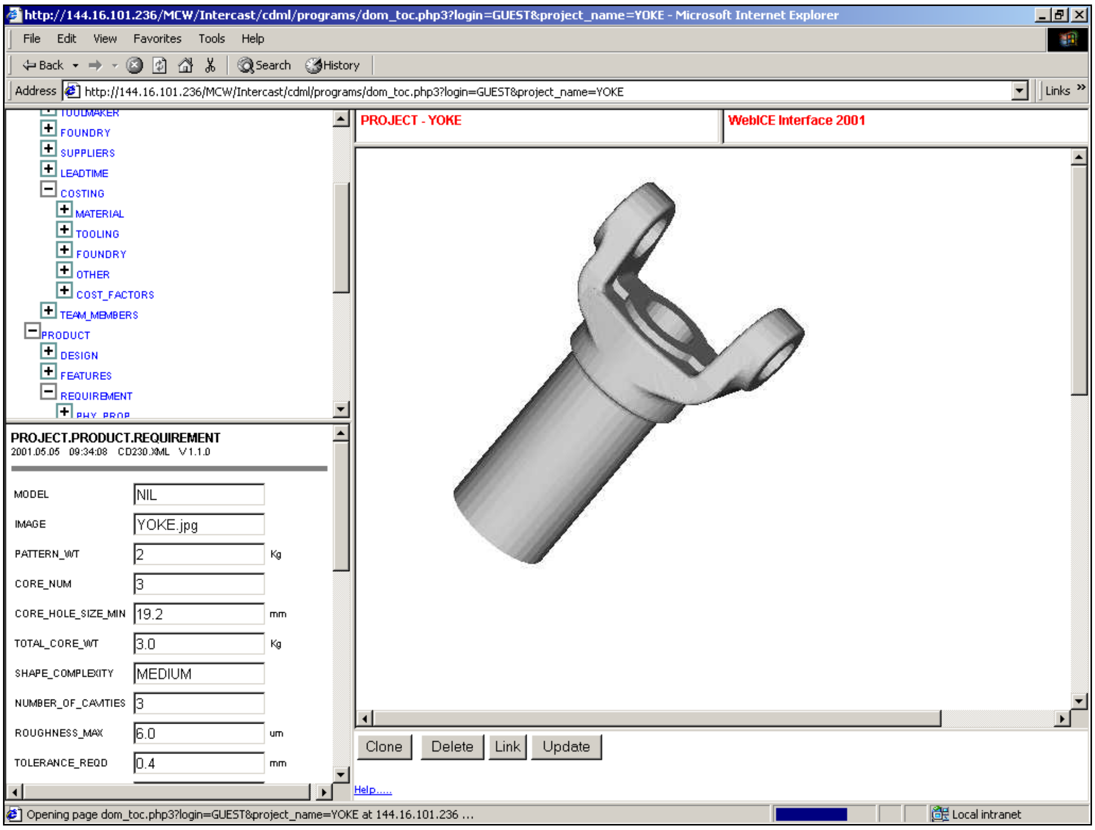


Figure 5.2 User interface

5.4 Functions

The various functions that have been included in the cost estimation software are ESTIMATE, DETAIL, SUMMARY, SAVE and HELP. These functions help the user in easy navigation through the cost estimation software. The action of each of these functions has been described below.

ESTIMATE: This particular function calculates the cost and displays them on the user interface. This function when called estimates the cost taking the default inputs and also displays them on the user interface in the form of a table.

DETAIL: This particular function when called gives the details of the cost estimation. It gives a detailed view of the various costs along with the input values that have been used.

SUMMARY: This particular function when called takes the user back to the first page where the original cost estimate information is displayed in a concise form.

COMPUTE: This particular function allows getting a new estimate after he has changed some data value for any of the cost factors.

SAVE: This particular function when called saves the estimated values in the corresponding database file and shows a message regarding the same on the interface.

HELP: This particular function when called gives tips on how to use the cost estimation application effectively, ensuring that all the input parameters are correctly specified so as to get accurate and consistent results.

5.5 Sample Session

The session begins with a user login and password validation at the site http://www.metalcastingworld.com/index.html. After that the user initiates either a new project or goes to an already created project and within the ADMIN block the user finds the cost block which when expanded while provide the interface with the cost buttons. After specifying the required inputs as described in the cost help screen on clicking the cost estimate button, the results are displayed on the screen.

A sample session of the cost estimation software was conducted for a ductile iron part 'yoke'. The inputs were to be given by the user by filling up the required data in the respective files. A help list is displayed when the user clicks the cost help button provided on the screen. The user interface showing the help list is shown in figure 5.3. The figure 5.4 shows the summarized results of the cost estimate when the user clicks the cost estimate button provided on the screen. The user either can **Save** the result or can view the **Details** of the cost estimation by clicking on the respective buttons provided on the user interface. The figure 5.5a-d show the details of the cost estimation results. The user can get a new estimate by changing any parameter by clicking on the **Calculate** button. Else, he can view the summarized results by clicking on the **Summary** button. The detailed view gives the list of all parameters used in the estimate, their values and their units. The user can get new estimate by changing the values of the parameters as per his need and then clicking on the **Calculate** button. This particular option facilitates the user to have estimates of the product by changing few features and thereby studying their effect on the cost of the product and thus helps him decide on the best alternative.

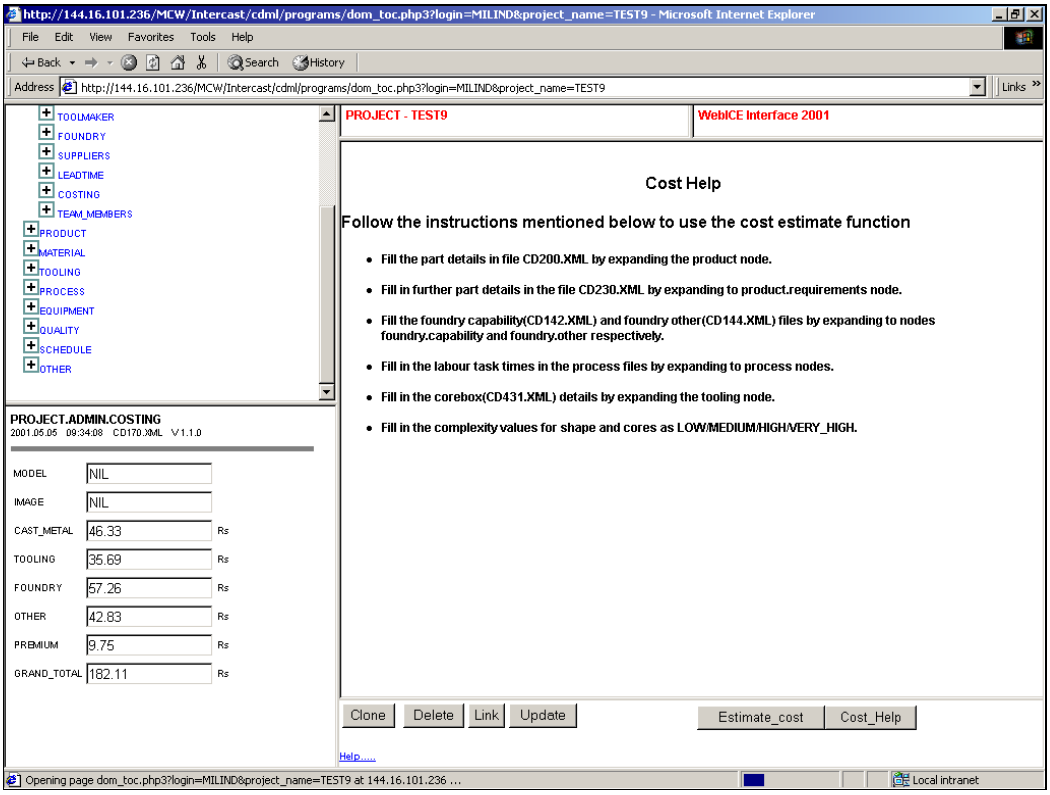


Figure 5.3 Cost help interface

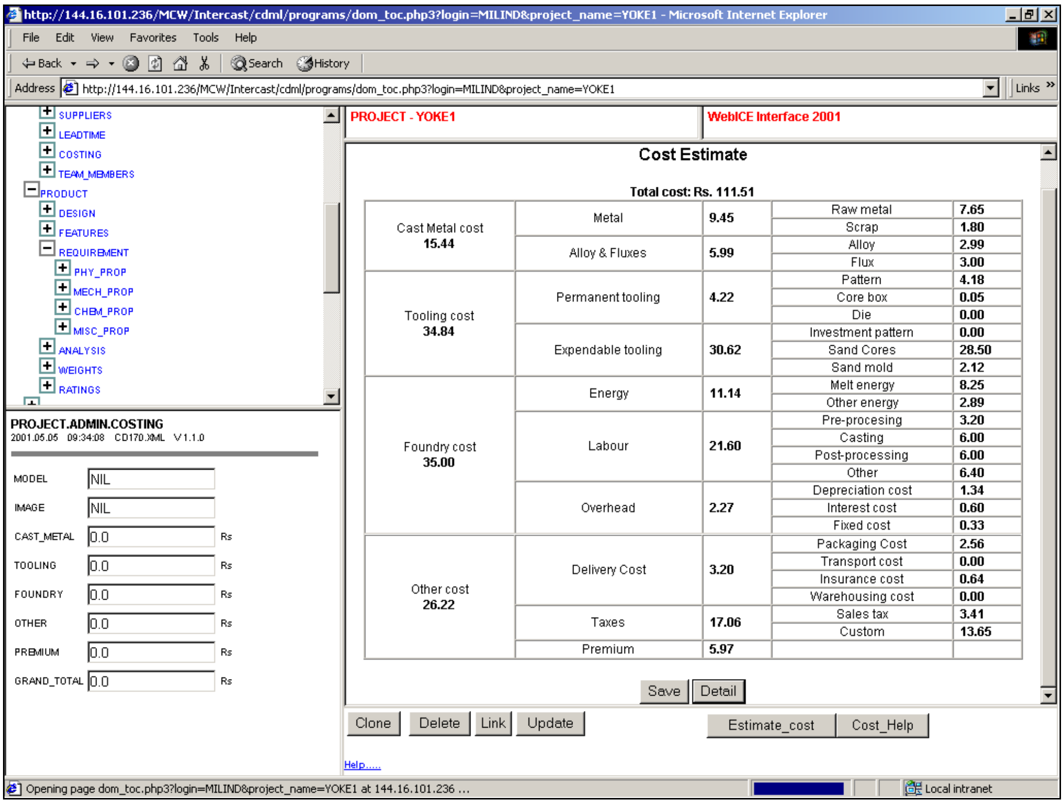


Figure 5.4 Cost estimate (Yoke)

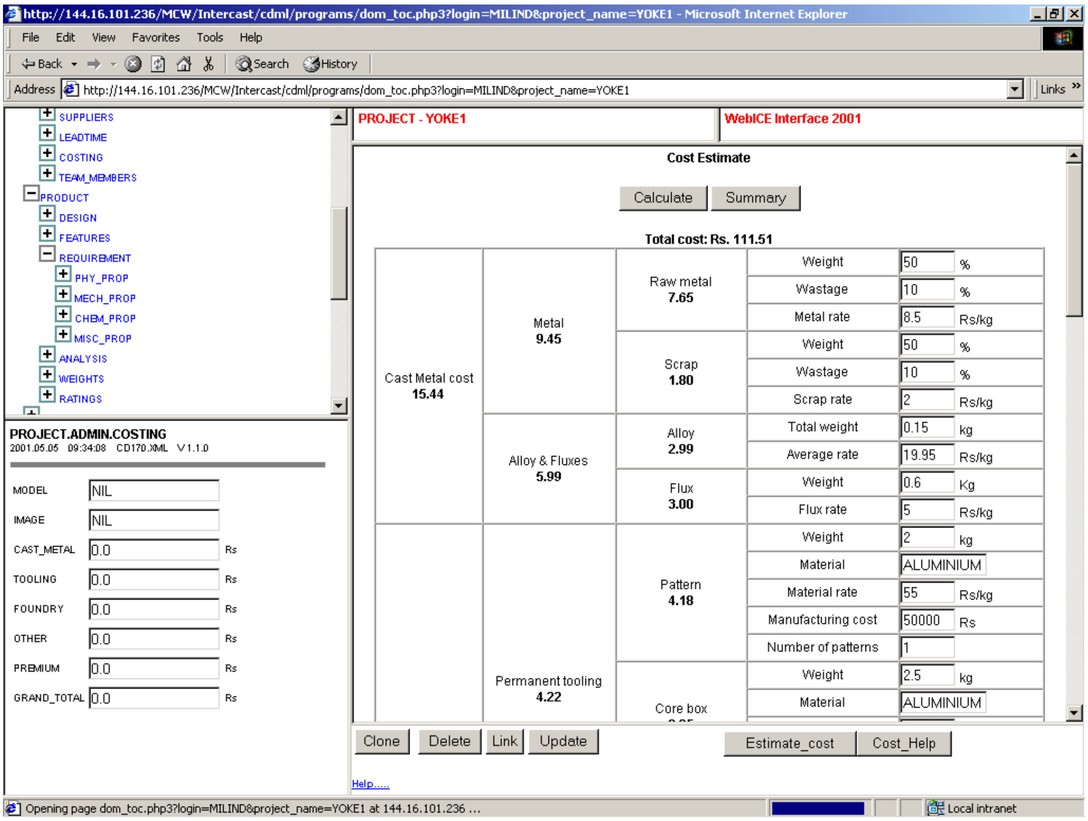


Figure 5.5a Cost detail (Yoke)

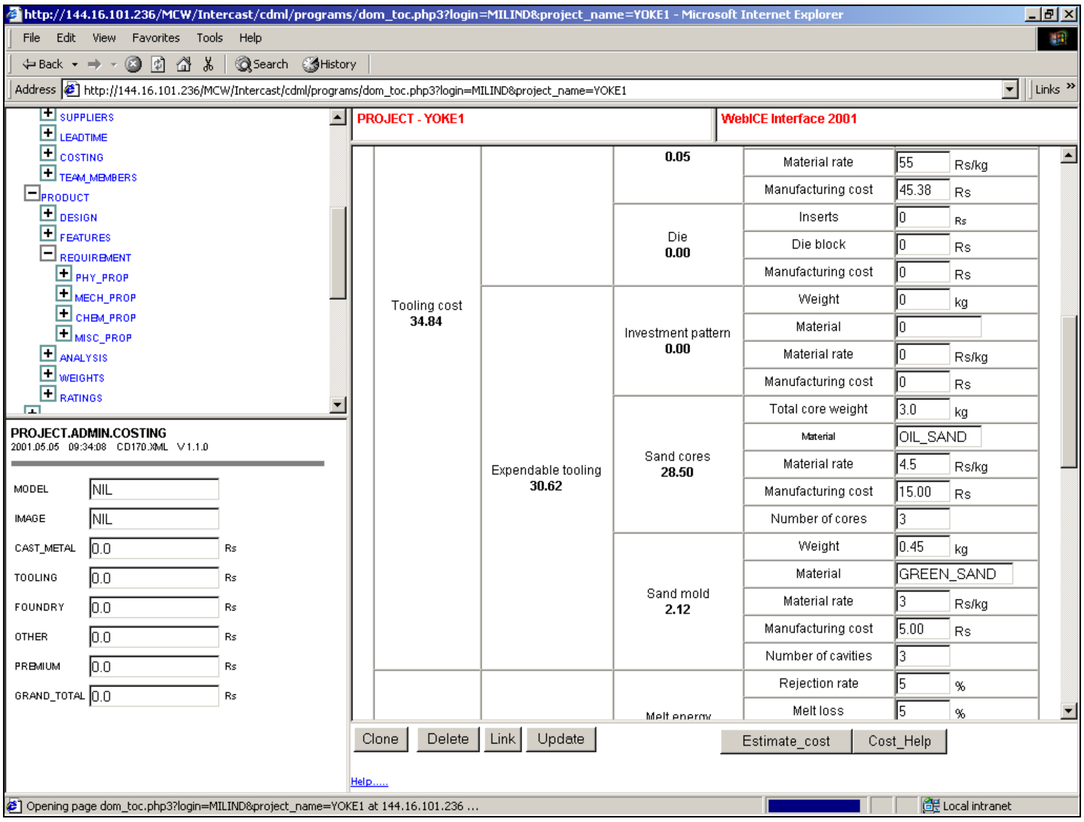


Figure 5.5b Cost detail (Yoke)

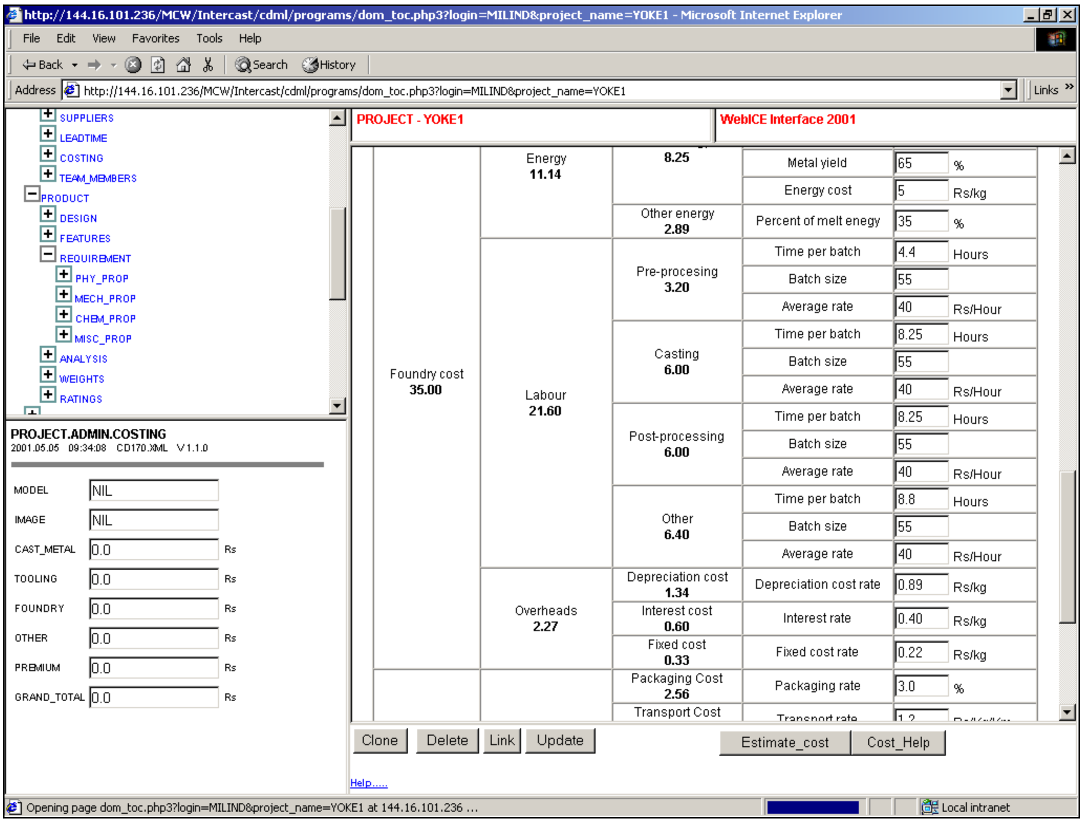


Figure 5.5c Cost detail (Yoke)

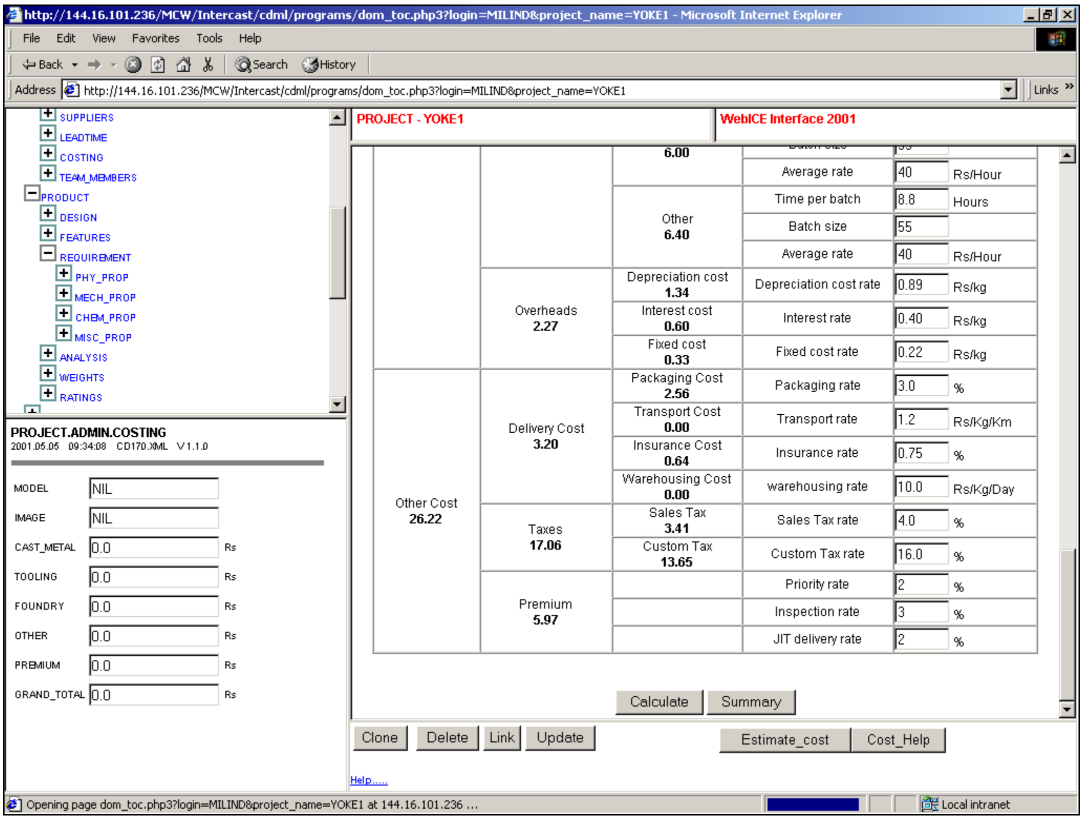


Figure 5.5d Cost detail (Yoke)

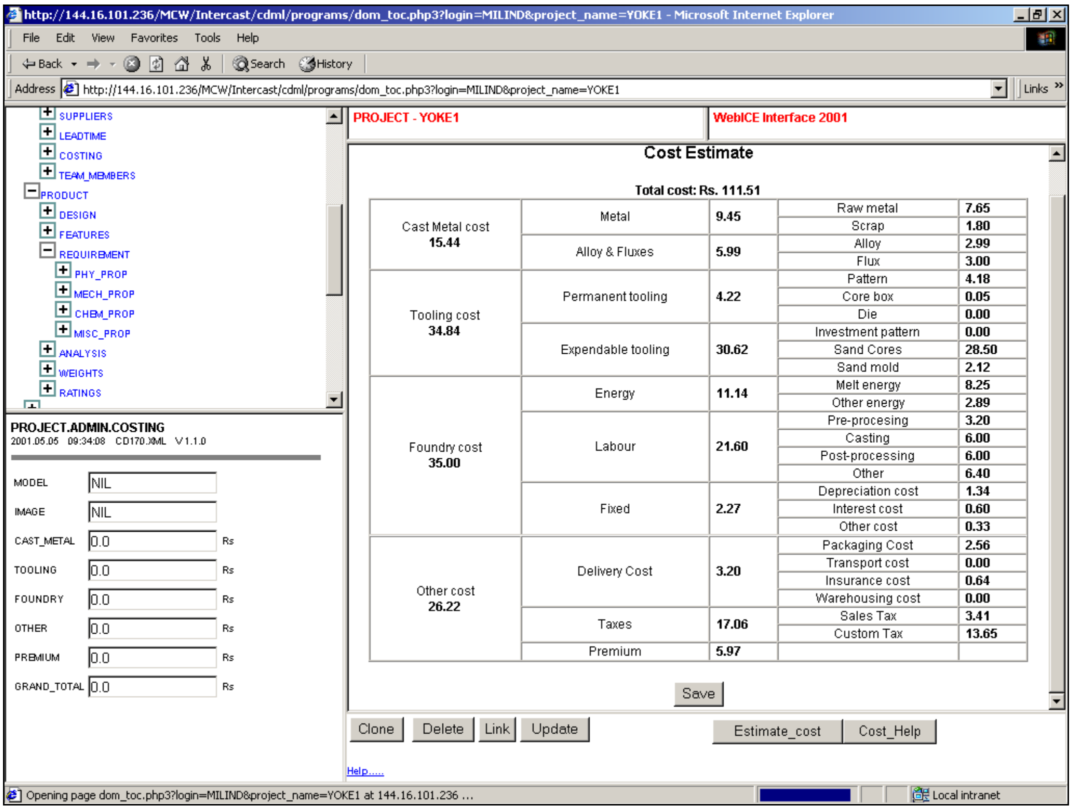


Figure 5.6 Cost summary (Yoke)

Figure 5.6 gives the summarized results of the cost estimate. Figure 5.7 shows the saved values in the cost file when the **Save** button is clicked.

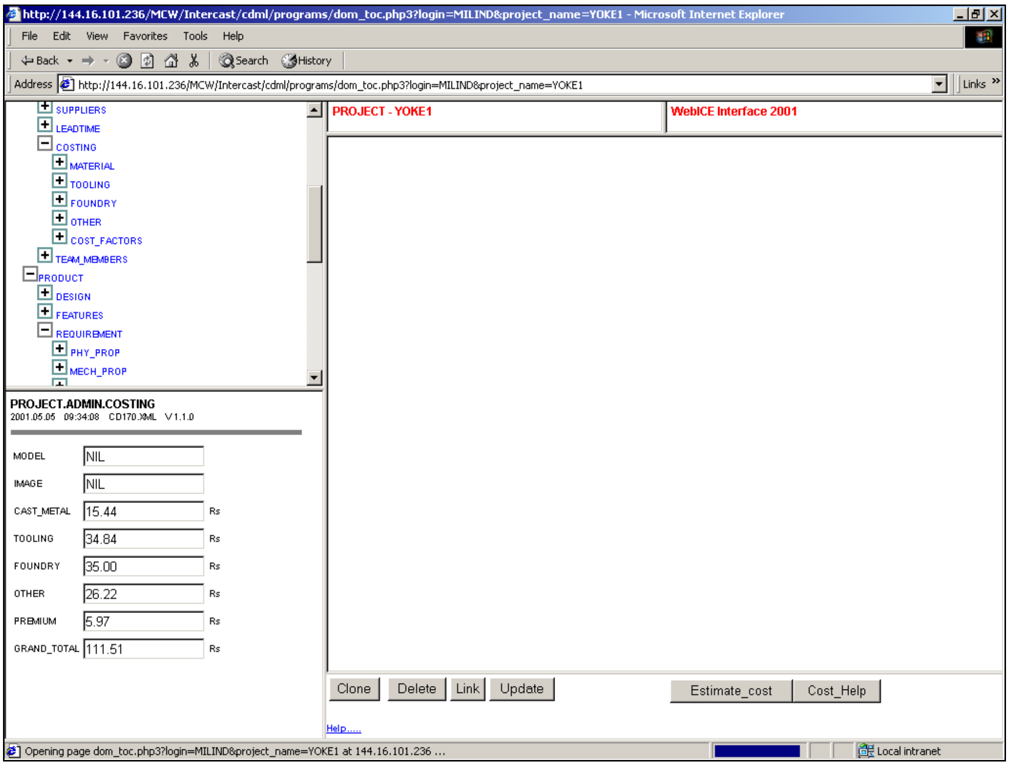


Figure 5.7 Cost save (Yoke)

One more sample session of an aluminium part 'compressor cover' has been demonstrated.

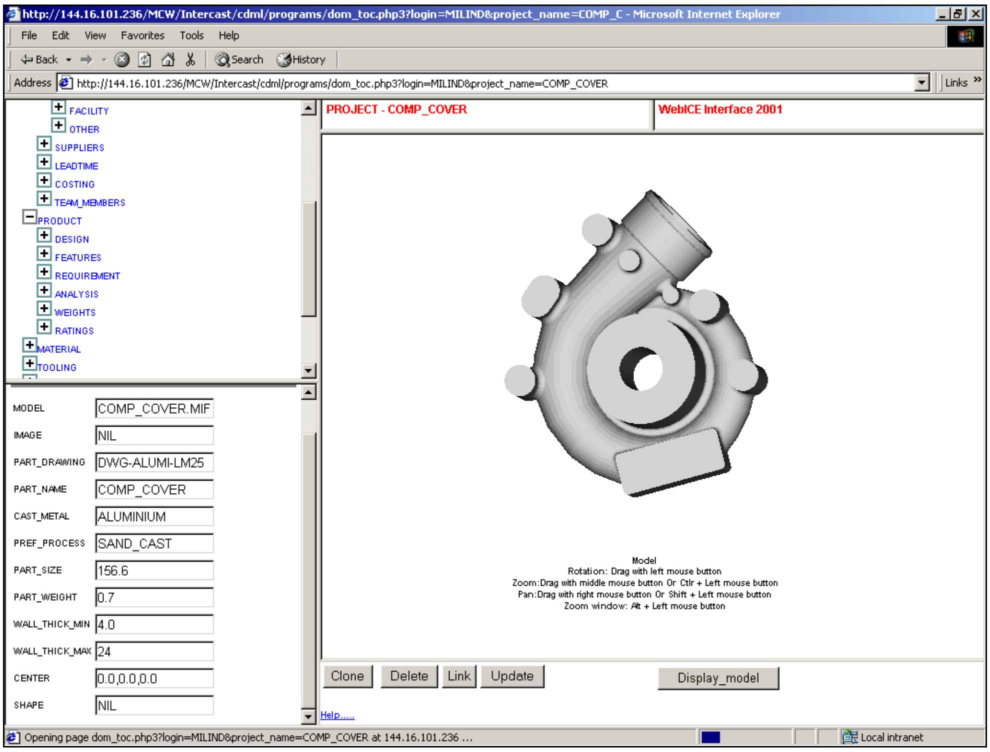


Figure 5.8 Compressor cover

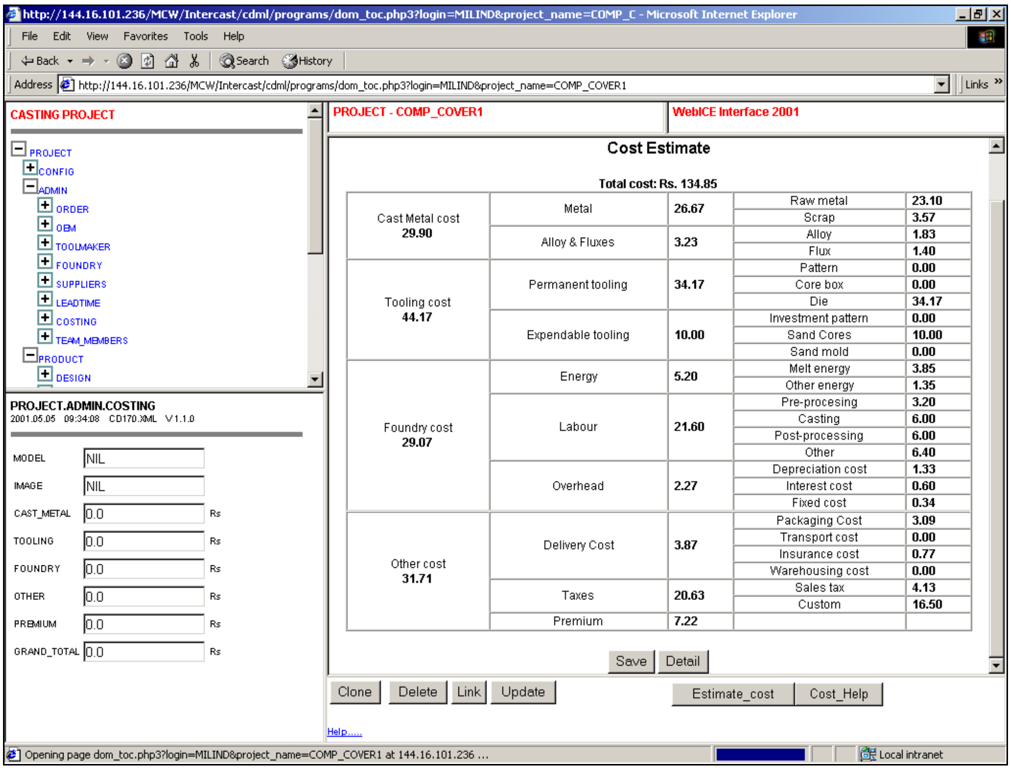


Figure 5.9 Cost estimate (Compressor cover)

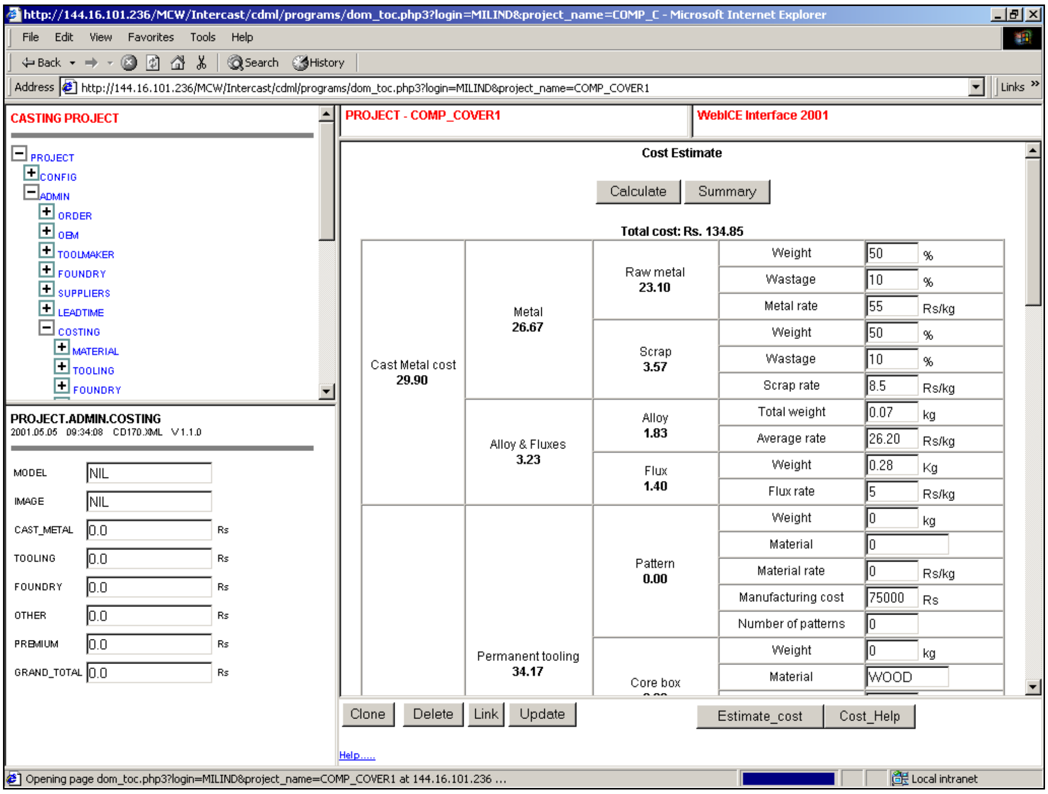


Figure 5.10a Cost detail (Compressor cover)

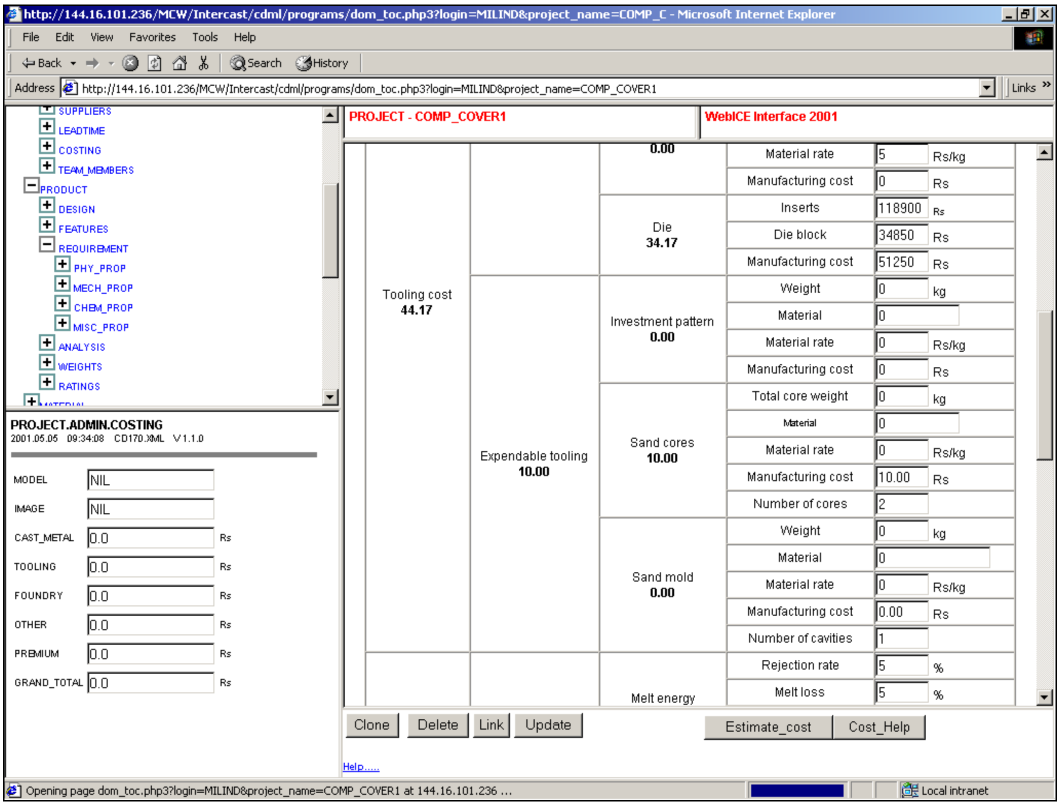


Figure 5.10b Cost detail (Compressor cover)

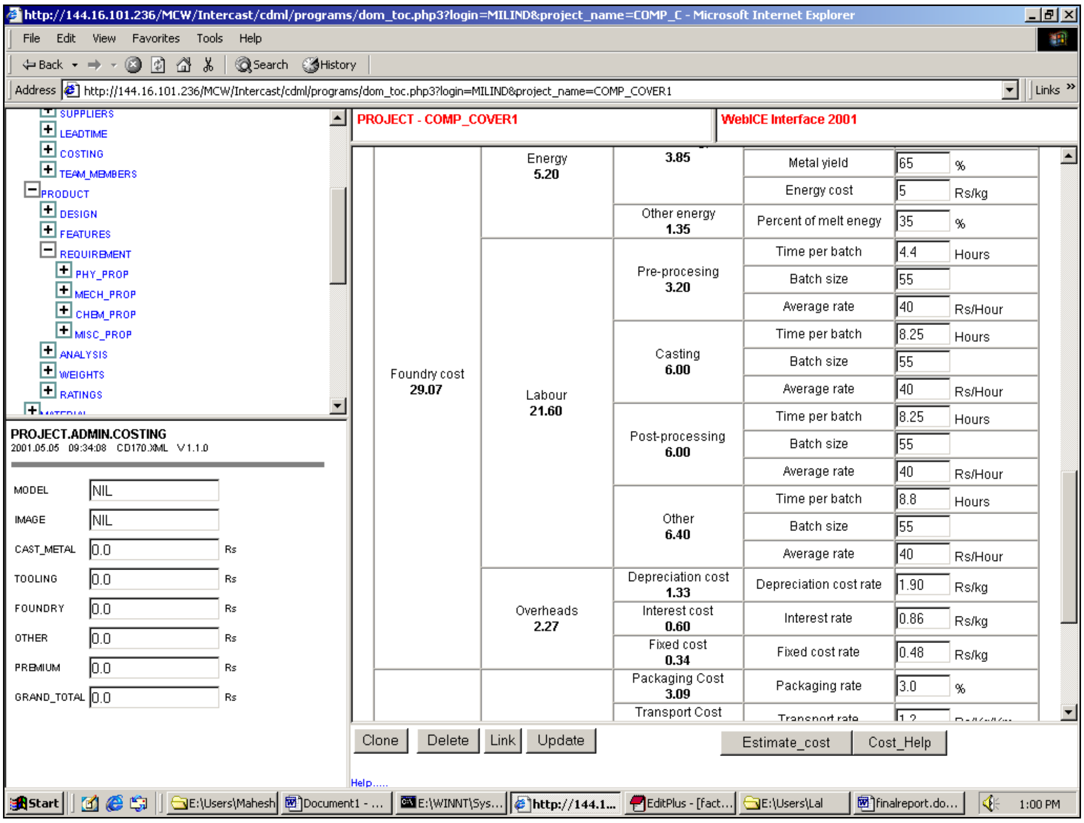


Figure 5.10c Cost detail (Compressor cover)

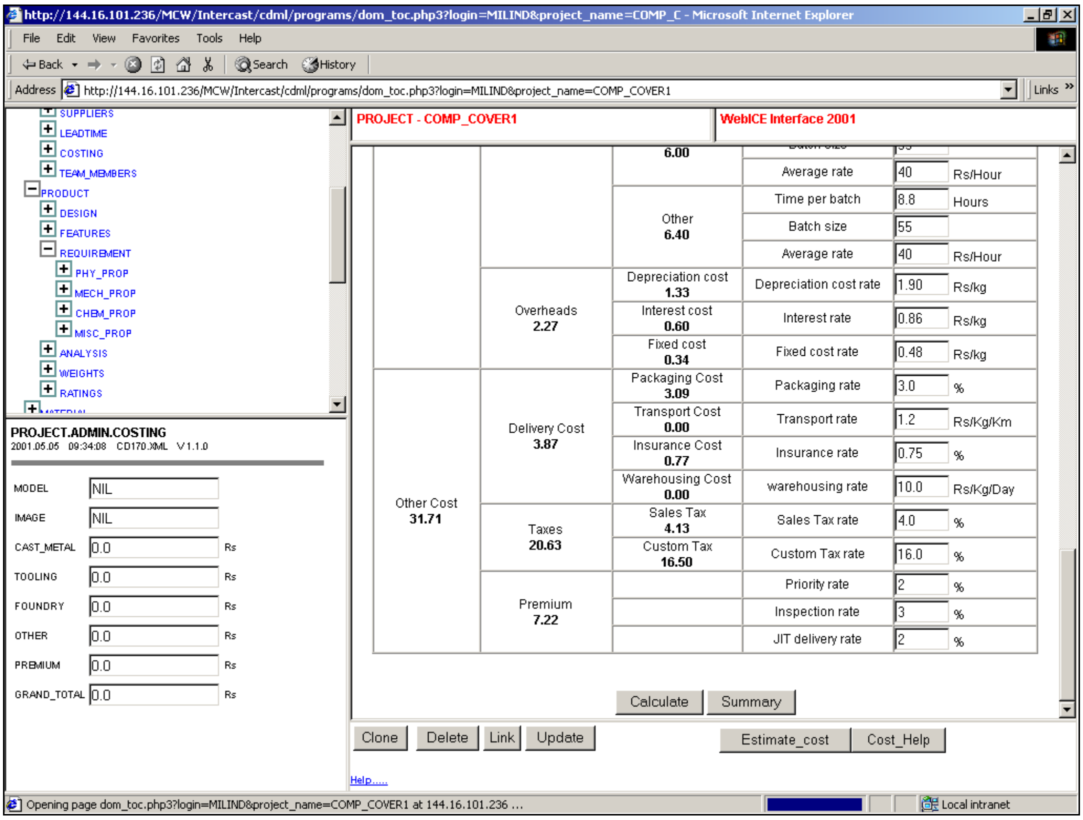


Figure 5.10d Cost detail (Compressor cover)

Thus, the program allows the user to get the cost estimate of a cast product by taking few pertinent inputs and displays the results of the estimate in a user-friendly interface. It also allows the user to study the effect of various features of the product on its cost, as he can get multiple estimates for the same product, by change the product features. This facility allows a designer to evaluate different design alternatives, at early design stage, from an economic point of view, which is very important for the success of the product in the market when it is launched. Further, since the results of the estimates can be saved and viewed at any point of time, provided the user has access rights to the data, this becomes very helpful tool in implementation of the collaborative engineering concept, thereby allowing the team-members to review the results of the estimates irrespective of their geographical location.

CHAPTER 6

CONCLUSIONS

6.1 Contributions

In today's highly competitive market, innovative and cost-effective design is the means to success, is a well-established fact. Since decisions taken at the design stage of a product influence its cost significantly, this stage becomes the most crucial one during the development of a product. With the aim being to design quality low cost products, the design-to-cost philosophy, wherein the product is designed to meet a predetermined target cost, is the right tool to achieve the same. Successful implementation of the design-to-cost philosophy requires tools to facilitate early cost estimation of the castings. Further, with the world growing into a global village and enterprises spreading their facilities across the globe in pursuit of the economies of scale, collaborative engineering is the new paradigm in the industry. In this project, an early cost estimation software as been developed, to facilitate the implementation of design-to-cost philosophy and the software has been web-enabled with the aim of facilitating collaborative engineering. The important contributions of this work are highlighted here.

Identifying Cost Factors:

Various factors affecting the casting process and thereby the cost of the product have been identified and their relationship with the cost of the cast product has been established.

Defining the Cost Block:

Based on the factors identified through the study of the casting process (sand casting, pressure die casting and investment casting), the various cost components of a product have been identified and the same have been modeled into a cost block, making use of CDML (an XML-compatible self-describing Casting Data Markup Language).

Implementing the Cost Model:

The cost model that has been developed, incorporating the important cost factors, has been successfully implemented in web-based collaborative engineering framework WebICE (Web-Based Integrated Collaborative Engineering).

Thus, an early cost estimation program for cast products has been developed and the use of same has been demonstrated in web-based collaborative engineering framework.

6.2 Limitations

The following are the few limitations of the work that has been done during the course of this project.

- The scope of the project has been limited to the field of castings and further more only three processes viz., sand casting, die-casting and investment casting have been considered, these being the most common and most widely applied processes.
- The software does not make any validation checks, so if the user enters wrong or inconsistent data the software might give erroneous results.
- The tooling cost, as of now is being calculated as a percentage of its material cost because of non-availability of sufficient data to perform regression analysis for the same.

6.3 Future Work

Further work can be done in the following areas of the project to increase the commercial utility of the project.

- A more detailed regression analysis could be conducted to determine the tooling cost, by collecting from different sources so that the equation so derived is consistent and generic in nature.
- Although all the pertinent cost factors have been included in the present model, new factors if identified and deemed necessary could be added to improve the accuracy of the software.
- A validation check with appropriate error and warning messages should be incorporated so that the user does not input incorrect or inconsistent data even by hindsight.
- The software could be extended to incorporate more processes to cater to the needs of larger clientele.

References

1. Akarte M. M., “A Framework for Web-Based Integrated Casting Engineering and its Application to Product-Process-Producer Compatibility Evaluation,” PhD Thesis, IIT Bombay, (2002).
2. Bharara A, Lee C. Y., “Implementation of an activity based costing system in a small Manufacturing Company,” *International Journal of Production Research*, Volume 34, No. 4, pp 1109-1130, (1996).
3. Bralla James G, “Hand Book for Product Design for Manufacturing,” McGraw-Hill, International Edition, pp 5.4-5.5, (1988).
4. AFS, “Casting Defect Analysis”, *http://207.250.195.21/4ebasic.htm*, last updated December (1999).
5. Chanan S. S., Unny M., “Concurrent Engineering: Concepts, Implementation and Practice,” Chapman & Hall, pp 11-25, (1994).
6. Cleaner Production Manual for the Queensland Foundry Industry, Nov. (1999).
7. Cooper R. and Chew B., “Control Tomorrow‘s Costs through Today’s Design,” *Harward Business Review*, pp 88-97, January-February (1996).
8. Cooper R. and Slagmulder R., “Target Costing and Value Engineering,” Productivity Press Portland, Oregon, (1997).
9. Creese R. C., Moore L. T., “Cost Modeling for Concurrent Engineering,” Cost Engineering, Volume 32, No. 6, pp 23-26, June (1990).
10. Creese R. C., Adithan M. and Pabla, B. S., “Estimating and Costing for the Metal Manufacturing Industry,” Cost Engineering Series, Volume 18, Marcel Dekker Inc., pp1-5, (1990).
11. Creese R. C., “Discreet Manufacturing Costs and Analysis,” Cost Engineering Series, Marcel Dekker Inc., pp 1-21, (1988).
12. Crookall J. R., “Design for Economic Manufacture,” *Annals of the CIRP*, Vol. 35, pp 93-97, Jan (1986).
13. Crow K., “Product Development Forum,” http://members.aol.com/drmassoc/dtc.htm, last updated September (2001).
14. Crow K., ”Product Development Forum,” http://members.aol.com/drmassoc/dfm.htm, last updated September (2001).

15. Dean U. B. and Unal R., “Designing for Cost,” *Transactions of the American Association of Cost Engineers*, 35th Annual Meeting, Seattle WA, June 23-26, pp D.4.1-D.4.6, (1991).
16. Distler H., Taether-Lordieck I. and Siemns A. G., “Developing Competitive Textile Machines on the basis of ‘Design to Cost,” Melliand English, pp 266-268, 11-12/ (1999).
17. Eias S. E. G., “Value Engineering A powerful Productivity Tool,” Computers Ind. Engng, Vol. 35, Nos. 3-4, pp 381-393, (1998).
18. Heine, R. W., and Rosenthal, P. C., “Principles of Metal Casting,” McGraw-Hill, New York, (1955).
19. Hundal M. S., “Rules for Models for Low Cost Design,” DE - volume 52, Design for Manufacturability, pp 75-84, ASME (1993).
20. Leibers, A. and Kals, H. J. J., “Cost Design Support in Product Design,” *Annals of the CIRP*, Volume 46, No. 1, pp 107-112, (1997).
21. Majumdar Nanda, “The New ABC of Cost Cutting,” Cover Story, *Business Today*, pp 63-75, Oct. 22- Nov. 6 (1997).
22. NASA, “Parametric Cost Estimating Handbook,” http://www.jsc.nasa.gov/bu2/ PCEHHTML/pceh.htm, last updated May (1999).
23. Nee A.Y.C., Ong S.K., Wang Y.G., “Computer applications in Near Net-Shape Operations,” London, Springer-Verlag, ISBN 1-85233-186-0, (1999).
24. Ong N. S., “Activity-based Cost Tables to Support Wire Harness Design”, *International Journal of Production Economics*, Vol. 29, pp 271-289, (1993).
25. Ravi, B., “Knowledge Based Casting Design”, 2nd World Foundry Congress, Philadelphia, USA, pp 23-26, April (1996a).
26. Ravi B., Akarte M. M., “Casting Information Management”, *AFS Transaction*, Volume 104, pp 217-223, (1996b).
27. Ravi, B., “Collaborative Design and Manufacture of Cast Products”, *Proceedings of the 45th Indian Foundry Congress*, pp 257-270, Jan (1997).
28. Sheldon D., Huang G., Perks R., “Specifications and Development of Cost Estimating Databases for Engineering Design,” DE- Vol. 52, Design for Manufacturability, ASME, pp 91-96, (1993).
29. Smith A., “Manufacturing advisory services,” http://cybercut.berkeley.edu/html/ research/Archive/Mas.html, last updated 4/20/ (1999).

30. Smith A., http://cybercut.berkeley.edu/mas2/html/applet/index.html, last updated 4/20/ (1999).

31. Spade A. T., “Casting Troubleshooting to be available via Internet,” *Modern Casting*, Vol. 88, No.2, pp38-40, (1998).

32. Swift, K. G. and Booker, J. D., “Process Selection”, John Wiley and Sons Inc., New York, (1997).

33. University of Maryland, “Workstations at Maryland,” http://www.wam.umd.edu/, last updated Oct (2001).

34. Vasvani, R., “Workshop on Foundry Trends”, *Indian Foundry Journal*, pp 48-49, Feb (1997).

35. Wen-Hsien Tsai, “Technical note on Work Sampling to estimate the Effort on Activities using Activity-based costing”, *International Journal of Production Economics*, Volume 43, pp 11-16, (1996).

Appendix

Description of CDML

First eight field names of each data block are constant. These are root node – CDML, VERSION, NAME, FILE, DATE, TIME, MODEL and IMAGE. The importance of these field names is given below.

The root node – CDML is an essential part of XML and it represents the parent of all information classified into a particular block.

The field name VERSION represents the current version of a particular data block. This can be used to check the incompatibility of a program for a particular version of a data block.

NAME field indicates the name of the data block, which represents three levels in the CDML tree hierarchy: child, its parent and grandparent, giving clear view of the information classification.

FILE gives the file name of the data block that can be used to either read from or modify the content of the file.

The field name DATE and TIME stores the date and time of data block creation or last update.

MODEL and IMAGE represents links to a 3D model and a 2D image respectively, which are stored in web browser compatible formats such as BMP, JPG and GIF for 2D and STL/VRML for 3D. For example, an EQUIPMENT block may have link to a picture of the equipment, and the PRODUCT block may have a link to a solid model of the part.

Full exploded view of the cost tree structure.

- PROJECT
  - CD001
  - CONFIG
  - ADMIN
    - CD100
    - ORDER
    - OEM
    - TOOLMAKER
    - FOUNDRY
    - SUPPLIERS
    - LEADTIME
    - COSTING
      - CD170
      - MATERIAL
        - CD171
      - TOOLING
        - CD172
      - FOUNDRY
        - CD173
        - ENERGY
        - LABOUR
        - OVERHEADS
      - OTHER
        - CD177
      - COST_FACTORS
        - CD178

The total cost (CD170.XML) file is shown in the following figure.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>               1.1.0                               </VERSION>
<NAME>                  PROJECT.ADMIN.COSTING               </NAME>
<FILE>                  CD170.XML                           </FILE>
<DATE>                  2001.05.05                          </DATE>
<TIME>                  09:34:08                            </TIME>
<MODEL CREATED="NO">        NIL                             </MODEL>
<IMAGE CREATED="NO">        NIL                             </IMAGE>
<CAST_METAL UNIT="Rs">              0.0                     </CAST_METAL>
<TOOLING UNIT="Rs">                 0.0                     </TOOLING>
<FOUNDRY UNIT="Rs">                 0.0                     </FOUNDRY>
<OTHER UNIT="Rs">                   0.0                     </OTHER>
<PREMIUM UNIT="Rs">                 0.0                     </PREMIUM>
<GRAND_TOTAL UNIT="Rs">             0.0                     </GRAND_TOTAL>

</CDML>
```

The material cost file (CD171.XML) is shown in the following figure.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.0                                   </VERSION>
<NAME>                 ADMIN.COSTING.MATERIAL                </NAME>
<FILE>                 CD171.XML                             </FILE>
<DATE>                 2001.05.05                            </DATE>
<TIME>                 09:34:08                              </TIME>
<MODEL CREATED="NO">        NIL                       </MODEL>
<IMAGE CREATED="NO">        NIL                       </IMAGE>
<BASIS>                              PART             </BASIS>
<CASTING UNIT="Rs">                  0.0              </CASTING>
<ALLOYING UNIT="Rs">                 0.0              </ALLOYING>
<TOTAL_CAST_METAL UNIT="Rs">         0.0              </TOTAL_CAST_METAL>
</CDML>
```


The following figure shows the tooling cost file (CD172.XML).


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.0                           </VERSION>
<NAME>                 ADMIN.COSTING.TOOLING         </NAME>
<FILE>                 CD172.XML                     </FILE>
<DATE>                 2001.05.05                    </DATE>
<TIME>                 09:34:08                      </TIME>
<MODEL CREATED="NO">        NIL                      </MODEL>
<IMAGE CREATED="NO">        NIL                      </IMAGE>
<PATTERN UNIT="Rs">                  0.0             </PATTERN>
<CORE_BOX UNIT="Rs">                 0.0             </CORE_BOX>
<DIE UNIT="Rs">                      0.0             </DIE>
<INV_PATTERN UNIT="Rs">              0.0             </INV_PATTERN>
<SAND_MOLD UNIT="Rs">                0.0             </SAND_MOLD>
<SAND_CORE UNIT="Rs">                0.0             </SAND_CORE>
<TOTAL_TOOLING UNIT="Rs">            0.0             </TOTAL_TOOLING>
</CDML>
```

The following figure shows the foundry cost file (CD173.XML).


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.0                                   </VERSION>
<NAME>                 ADMIN.COSTING.FOUNDRY                 </NAME>
<FILE>                 CD173.XML                             </FILE>
<DATE>                 2001.05.05                            </DATE>
<TIME>                 09:34:08                              </TIME>
<MODEL CREATED="NO">             NIL                         </MODEL>
<IMAGE CREATED="NO">             NIL                         </IMAGE>
<TOTAL_ENERGY UNIT="Rs">         0.0           </TOTAL_ENERGY>
<TOTAL_LABOUR UNIT="Rs">         0.0           </TOTAL_LABOUR>
<TOTAL_OVERHEADS UNIT="Rs">      0.0           </TOTAL_OVERHEADS>
<TOTAL_FOUNDRY UNIT="Rs">        0.0           </TOTAL_FOUNDRY>
</CDML>
```


The following figure shows the energy cost file (CD174.XML).


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.0                                   </VERSION>
<NAME>                 COSTING.FOUNDRY.ENERGY                </NAME>
<FILE>                 CD174.XML                             </FILE>
<DATE>                 2001.05.05                            </DATE>
<TIME>                 09:34:08                              </TIME>
<MODEL CREATED="NO">             NIL                         </MODEL>
<IMAGE CREATED="NO">             NIL                         </IMAGE>
<BASIS>                          PART                        </BASIS>
<ELEC_POWER_RATE UNIT="Rs/kw-hr">   5.0   </ELEC_POWER_RATE>
<MELTING UNIT="Rs">                 0.0   </MELTING>
<OTHER_ENERGY UNIT="Rs">            0.0   </OTHER_ENERGY>
<TOTAL_ENERGY UNIT="Rs">            0.0   </TOTAL_ENERGY>
</CDML>
```


The following figure shows the labour cost file CD175.XML.


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
        <VERSION>              1.0                           </VERSION>
        <NAME>                 COSTING.FOUNDRY.LABOUR        </NAME>
        <FILE>                 CD175.XML                     </FILE>
        <DATE>                 2001.05.05                    </DATE>
        <TIME>                 09:34:08                      </TIME>
        <MODEL CREATED="NO">        NIL                      </MODEL>
        <IMAGE CREATED="NO">        NIL                      </IMAGE>
        <BASIS>                     PART                     </BASIS>
        <LABOUR_RATE UNIT="Rs/hr">  0.0     </LABOUR_RATE>
        <PRE_CASTING UNIT="Rs">     0.0     </PRE_CASTING>
        <CASTING_COST UNIT="Rs">    0.0     </CASTING_COST>
        <POST_CASTING UNIT="Rs">    0.0     </POST_CASTING>
        <OTHER_COST UNIT="Rs">      0.0     </OTHER_COST>
        <TOTAL_LABOUR UNIT="Rs">    0.0     </TOTAL_LABOUR>
</CDML>
```

The following figure shows the overhead cost file (CD176.XML).


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
        <VERSION>          1.0                          </VERSION>
        <NAME>             COSTING.FOUNDRY.OVERHEAD</NAME>
        <FILE>        CD176.XML                    </FILE>
        <DATE>             2001.05.05                   </DATE>
        <TIME>        09:34:08                </TIME>
        <MODEL CREATED="NO">     NIL                   </MODEL>
        <IMAGE CREATED="NO">     NIL                   </IMAGE>
        <BASIS>                       YEAR              </BASIS>
<DEPRICIATION_RATE UNIT="Rs/Kg"> 0.0      </DEPRICIATION_RATE>
<DEPRICIATION_COST UNIT="Rs">    0.0      </DEPRICIATION_COST>
<INTEREST_RATE UNIT="Rs/Kg">     0.0      </INTEREST_RATE>
<INTEREST_COST UNIT="Rs">        0.0      </INTEREST_COST>
<FIXED_RATE UNIT="Rs/Kg">        0.0      </FIXED_RATE>
<FIXED_COST UNIT="Rs/Kg">        0.0      </FIXED_COST>
<TOTAL_OVERHEADS UNIT="Rs">      0.0      </TOTAL_OVERHEADS>
</CDML>
```


The following figure shows the other cost file (CD177.XML)


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
        <VERSION>          1.0                          </VERSION>
        <NAME>             ADMIN.COSTING.OTHER          </NAME>
        <FILE>        CD177.XML                    </FILE>
        <DATE>             2001.05.05                   </DATE>
        <TIME>        09:34:08                </TIME>
        <MODEL CREATED="NO">     NIL                   </MODEL>
        <IMAGE CREATED="NO">     NIL                   </IMAGE>
        <BASIS>                       BATCH        </BASIS>
        <PACKAGING UNIT="Rs">     0.0          </PACKAGING>
        <TRANSPORT UNIT="Rs">     0.0          </TRANSPORT>
        <INSURANCE UNIT="Rs">          0.0          </INSURANCE>
        <WAREHOUSE UNIT="Rs">     0.0          </WAREHOUSE>
        <TOTAL_DELIVERY UNIT="Rs">0.0          </TOTAL_DELIVERY>
        <SALES_TAX UNIT="Rs">          0.0          </SALES_TAX>
        <CUSTOM UNIT="Rs">        0.0          </CUSTOM>
        <TOTAL_TAXES UNIT="Rs">   0.0          </TOTAL_TAXES>
        <PREMIUM UNIT="Rs">            0.0          </PREMIUM>
        <TOTAL_OTHER UNIT="Rs">    0.0         </TOTAL_OTHER>
</CDML>
```

The following figure shows the cost factors file (CD178.XML)


```
<?xml version="1.0" ?>
<CDML DOMAIN="CASTING DATA MARKUP LANGUAGE">
<VERSION>              1.0                                  </VERSION>
<NAME>                 ADMIN.COSTING.COST_FACTORS           </NAME>
<FILE>                 CD178.XML                            </FILE>
<DATE>                 2001.05.05                           </DATE>
<TIME>                 09:34:08                             </TIME>
<MODEL CREATED="NO">       NIL                              </MODEL>
<IMAGE CREATED="NO">       NIL                              </IMAGE>
<PACKAGING_RATE UNIT="%">                3.0     </PACKAGING_RATE>
<TRANSPORT_RATE UNIT="Rs/Kg/Km">         1.2     </TRANSPORT_RATE>
<INSURANCE_RATE UNIT="%">                0.75    </INSURANCE_RATE>
<SALES_TAX_RATE UNIT="%">                4.0     </SALES_TAX_RATE>
<CUSTOM_RATE UNIT="%">                   16.0    </CUSTOM_RATE>
<WAREHOUSING_RATE UNIT="Rs/Kg/Day">10.0</WAREHOUSING_RATE>
</CDML>
```