RE-ENGINEERING THE PROPOSAL PROCESS USING PARAMETRIC COST MODELS

by

Linda G. Berrey

Major Paper submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Systems Engineering

APPROVED:

Kostas P. Triantis, Chairman

Benjamin S. Blanchard

Donald W. Thomson

May, 1992

Blacksburg, Virginia

RE-ENGINEERING THE PROPOSAL PROCESS
USING PARAMETRIC COST MODELS

by

Linda G. Berrey

Committee Chairman: K. Triantis
Industrial Systems Engineering

(ABSTRACT)

Parametric cost models are powerful tools that can be beneficial throughout the three major areas of a development program: planning, estimating, and control. Parametric cost models can be used to generate program cost estimates at a conceptual level and support a bid/no-bid decision in the planning stages of a proposal. They can be used to estimate program costs at a detailed level for submission in a proposal or help substantiate estimates derived by other methods. They can also be used as a measurement tool to improve quality, control costs, and identify risks during contract performance.

Processes that exist for these cost-related activities at IBM Federal Systems Company (FSC) in Manassas, Virginia are discussed in terms of their limitations. The use of parametric cost analysis can supplement the current processes and provide structure and objectivity where little

are evident. It can also reduce time, cost, errors, and labor involved in performing these activities. The result of the paper is a *re-engineered* proposal process incorporating the use of parametric cost models.

This paper presents a conceptual discussion. Limited implementation is described in terms of an example program, Beta, where a parametric cost estimating method is tested. Based on the results of this example, a quantitative evaluation is made. Further plans for validating the proposed solution and justifying its use are outlined. Common concerns and objections about parametric cost models are addressed. The Parametric Review of Information for Costing and Evaluation of Hardware (PRICE-H) model is used as a case study to illustrate some of the possible applications of parametric models and how they are performed. Specific limitations of the PRICE model are discussed.

Table of Contents

Figures

Tables

I. Introduction

Reducing costs and improving quality is recognized as an important endeavor, especially in the arena of national defense where budget constraints are increasing. "The President's Blue Ribbon Commission report on defense in 1986 concluded the Department of Defense's procurement process for complex systems was costly, time consuming, labor intensive and error prone."[1] Experience has shown that this is also true of the process through which contractors develop proposals.

A proposal is a legally binding offer to provide goods and services and is the necessary foundation for contract performance. For this paper, the proposal process is viewed from the perspective of the supplier of systems and subsystems to a Federal customer, specifically the Department of Defense (DoD). Although many of the concepts presented apply to contractors and customers in the private sector, discussion is limited to Federal defense systems. The specific application addressed here is the cost estimating group who supports the proposal manager. Figure 1 illustrates the relationship between the supplier proposal process and the activities involved in the Federal procurement process.

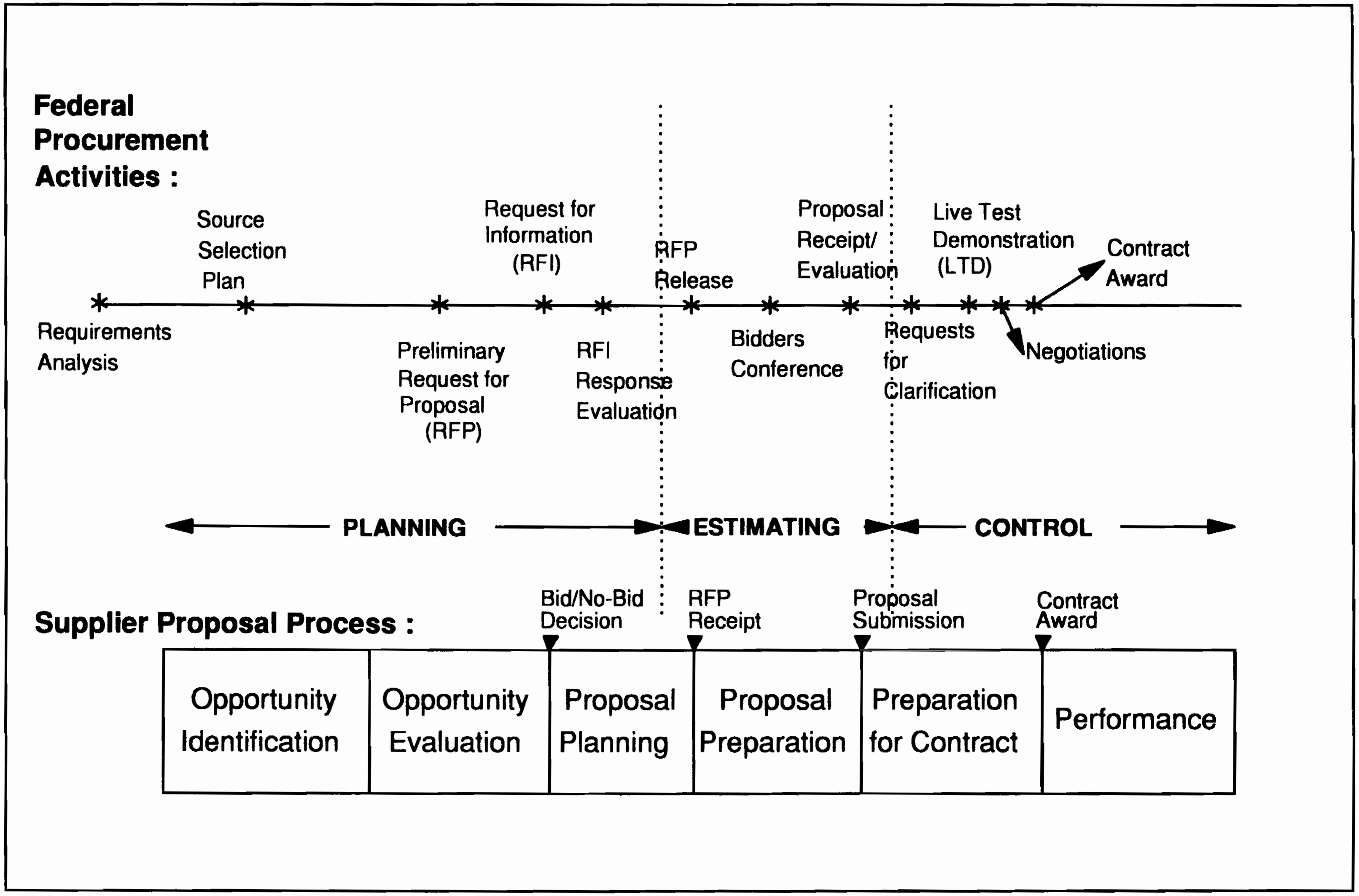


Figure 1 - Federal Procurement Process & Supplier Proposal Process

The proposal life cycle can be divided into three phases beginning with the identification and evaluation of a business opportunity. It is at this time when the customer and the competition are analyzed to determine possible customer funding and the probability of a win. Based on this analysis, a bid/no-bid decision is made and priority for this proposal among other potential projects is established. Systems engineering activities, including the development of a conceptual baseline, fall into this initial phase. Other planning activities, such as scheduling, staffing, and generating Bid and Proposal (B&P) (that is, funds set aside specifically for pursuing new business) or Independent Research and Development (IR&D) spending requirements, are also generally completed prior to receipt of the formal request for proposal (RFP) from the customer.

The second phase begins with receipt of the formal RFP and ends with submission of the proposal to the customer. During this phase, a proposal plan is developed along with schedules to meet it. A kick-off meeting involving all section authors and team members who will submit estimates is held. Finally, the proposal volumes (typically technical, cost, and management) are prepared and submitted. The preparation of the proposal is the most intensive phase of the process, marked by peak staffing and B&P expenditures and severe time constraints.

The third phase begins after the proposal submission and, for a winning proposal, extends into contract performance. Prior to contract award, responses may be made to customer requests for clarifications and best and final offer (BAFO) may be submitted. Also, if required, a live test demonstration (LTD) is conducted, followed by factfind (the process through which auditors ask questions and receive additional substantiation for proposal inputs) and negotiations. The final step is contract performance which may continue for years. Performance in terms of technical attributes, cost, and schedule are monitored and controlled in accordance with the terms and conditions of the contract. This proposal cycle is tightly coupled with the Federal customer's procurement activities as shown in Figure 1.

Each year, industry spends millions of dollars of B&P expense responding to customer RFPs. Figure 2 illustrates a detailed breakout of a typical proposal process. The technical team is made up of engineers who develop a solution to the customer's problem and who will complete the design once the contract is awarded.

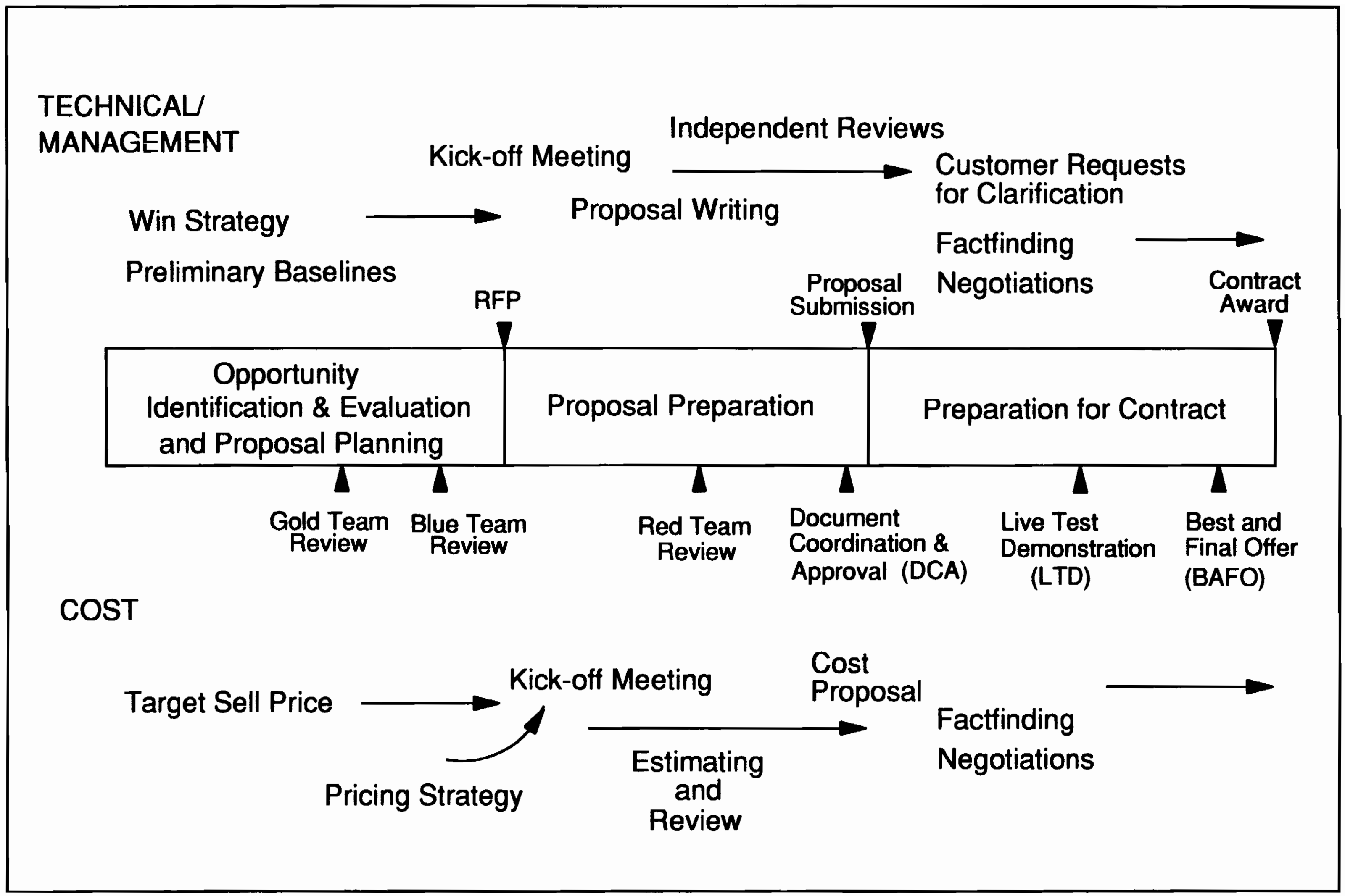


Figure 2 - Supplier Proposal Process

The management proposal is prepared by the person or persons who will manage and control the performance on the contract. Proposal leaders develop a *win strategy* and format for proposal writing. Generally, there is a limit to the number of pages allowed in the technical and management proposal volumes; accurate and succinct writing is crucial.

Independent reviews are conducted to ensure clarity and completeness of the proposal content, as well as the adequacy of the system solution. The objective of the initial review is to assess compliance and competitiveness of the preliminary system design. The next review addresses the proposal team preparedness, including *win strategies*, themes, author templates, proposal staffing and assignments, as well as bid strategies. Other teams review the proposal draft at two different stages to ensure compliance with the statement of work (SOW), proposal preparation instructions, and evaluation criteria, and to assess overall quality, clarity of presentation, and persuasiveness. Finally, Document Coordination and Approval (DCA) is the executive management review of the proposal prior to submission.

The cost team for a hardware proposal is comprised of a representative from each of the organizations that will perform on the contract and one or more cost engineers. Each representative from the performing organizations submits a budget sheet with supporting data that details the amount of effort required for the activities that his or her area will complete. In addition, the cost engineer is responsible for preparing an estimate and accompanying support for all recurring (production) hardware costs and hands-on labor associated with it. The amount of documentation in the cost proposal is generally not limited by the customer and can be astronomical.

A typical RFP allows the contractor between 30 and 90 days to submit a proposal. In many cases, work is begun by the technical team prior to receipt of the RFP and prior to the involvement of the cost team. The technical team chooses an architecture solution and begins work on the technical proposal. At this point, proposal leaders have no visibility (insight) into the total cost estimate and therefore no feeling for its adequacy. Often the result is a cycle of revising budget estimates if it is determined that the cost is unreasonable and rewriting the technical volume to correspond to these new cost inputs. Due to tight proposal deadlines, little time remains to make these revisions for timely proposal submission.

The more complex systems require the concentrated efforts of dozens and possibly hundreds of people to prepare a proposal. In any development effort, as more people become involved, the number of communication channels increases, and the opportunity for error increases in turn. For instance, if a change to the system architecture is made once the cost estimating process begins, all members of the cost and technical teams who may be affected must be aware of the change. All budget sheets must reflect the same ground rules and assumptions.

This paper addresses problems that have been identified in the existing proposal process. Lengthy proposal preparation cycles, high costs, involvement of large numbers of people, opportunities for error due to ineffective communication, and lack of visibility into program cost estimates are some of these problems that the proposed solution attempts to correct. In addition, the paper suggests a standard method for influencing bid/no-bid decisions, conducting technical trade studies related to cost, developing vendor *should-costs* for negotiation support, assessing quality, establishing risk areas, preparing cost estimates for proposal submission, and validating cost estimates performed by other methods.

The paper proceeds with a description of the current process for each area (planning, estimating, and control), with a focus on cost estimating methods. Parametric cost models are introduced as a potential solution to the problems identified, followed by a description of their application to each of the major areas. Advantages and disadvantages of the parametric approach are presented. A plan for validating the proposed method and justifying its use is outlined, including an example program. Finally, conclusions of the research and recommendations for its use are made. Throughout the paper, the PRICE model is used as a case study to demonstrate some of the suggested applications of parametric cost models. This does not imply that PRICE is the preferred tool over other available parametric models. Instead, the solution focuses on the concept of parametric cost models.

This study is based on the proposal process of a contractor to the DoD. Stringent regulations imposed by the Federal government must be considered. The solution is viewed from the standpoint of the group who performs the cost estimating function in support of the proposal process.

II. Planning

The planning aspect of a proposal involves opportunity identification and evaluation and other activities included in the systems engineering process.

Systems Engineering Process

As stated in Military Standard 499A, in the DoD context, the purpose of systems engineering is threefold. First, it is an effort "to transform an operational need into a description of system performance parameters and a system configuration through the use of an iterative process of definition, synthesis, analysis, design, test, and evaluation." Second, it is intended to "integrate related technical parameters and ensure compatibility of all physical, functional, and program interfaces in a manner that optimizes the total system definition." And third, it is the application of scientific and engineering efforts to "integrate reliability, maintainability, safety, survivability, human, and other such factors into a total engineering effort to meet cost, schedule, supportability and technical performance objectives." "In its simplest terms, systems engineering is both a technical process and a management process."[2]

As part of the management process, it is the responsibility of the systems engineer to understand all interfaces and to integrate the efforts of all engineering specialties to form a single solution.

For instance, a system must be created taking the user into consideration; there is no benefit in a system that cannot be operated by the intended user. This is known as human factors or human engineering. The systems engineer must also understand the reliability and performance requirements of the system. A system that fails after ten minutes of use is of no value unless ten minutes is the required life of the system. This is where the reliability engineer becomes involved. The maintainability engineer must assure that the system can be operated and maintained whether it be by the user or by repair technicians. The design must reflect criticality and accessibility for repair.

Integrated logistics support (ILS) is also important for decisions regarding the life cycle. According to the government, the system life cycle is the "period of time between the establishment of an operational need and the removal of a system from service."[3] This includes concept exploration, demonstration and validation, full-scale development, production and development, operations and maintenance, and disposal. The customer is interested not only in the acquisition cost of a system, but in the cost of operating and maintaining that system throughout its required life as well.

Hardware engineering, component engineering, software engineering, test engineering, mechanical engineering and any other detailed technical expertise required to design a system that meets the customer's operational needs are employed to formulate a producible and supportable solution. There is a critical need to involve other technical disciplines at an early stage instead of waiting until problems occur. At that point it may be too late to impact the design without adversely affecting the cost or the schedule. The systems engineer's role is to define the requirements for a solution to the customer's needs as set forth in the RFP, SOW, and customer meetings, provide leadership and coordination of all engineering specialties comprising the design team, direct trade studies between the disciplines, and assure that the system meets design requirements.

Systems engineering as a technical process can be divided into a number of primary activities, as illustrated in Figure 3. The process begins with an analysis of system requirements. This is considered one of the most important activities as this is where a baseline is established. If customer requirements are not fully understood at the outset, the right solution to the wrong problem may be created.

After a baseline is established, the system architecture is developed based on requirement specifications. This is an iterative (repetitive), top-down process starting at a conceptual level and evolving toward specific hardware and software designs. Trade studies are conducted to identify strengths and weaknesses of possible alternatives and provide visibility into risks.

After an architecture is chosen, a detailed design of system hardware and software components is completed. The system is integrated, tested, and installed at the location, and the end-user is trained. Maintenance, operation, and evaluation are ongoing activities throughout the life of the system that are supported by the system engineer.

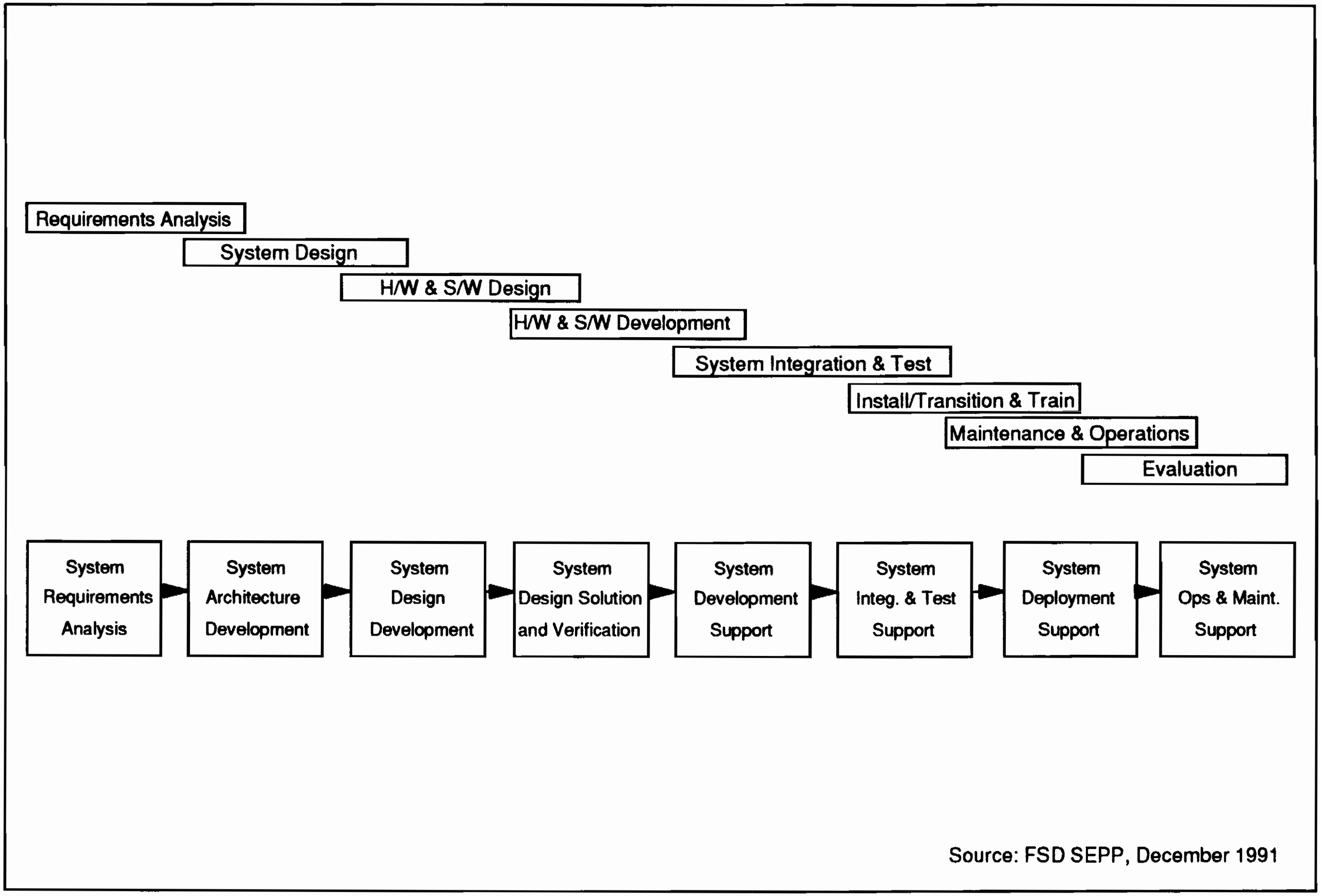


Figure 3 - Systems Engineering Process

According to *Systems Engineering Principles and Practices*, [Systems engineering is defined in FSC as]..."the iterative controlled process in which users needs are understood and evolved, through incremental development of requirements specifications and system design, to an operational system."[4]

Emphasis is placed on systems engineering as an iterative process. A design concept is formulated from initial meetings with the customer to understand the operational needs and system requirements. As more information concerning requirements, architecture alternatives, parts and material availability, reliability, and so forth, becomes available, the specifications change and the design concept changes. This is an important point to consider. The Blue Ribbon Commission report also stated that "systems are implemented as if they had been completely pre-specified although experience has shown repeatedly that specifications undergo considerable modification during development."[5] These modifications are often the result of changes in customer requirements and not necessarily due to errors made in specifications.

Architecture Selection

The system architecture development is a subprocess of the total systems engineering effort and is the first level of design, created top-down starting with a concept. As illustrated in Figure 4, five major tasks have been identified in completing this phase of the systems engineering process.[6]

The first task is the analysis of architecture requirements presented in the RFP from the customer. This document also provides the requirements-based trade-off criteria. The standard set of criteria should include: system cost, contract development and delivery schedule, system environment, testability of each requirement, and risks. The goal is to provide early insight into what constitutes a valid system architecture addressing all explicit, implied, and derived user requirements.

The second task is to develop and document the top-level functional architecture, followed by task three which includes the development of candidate architectures using application and development experience.

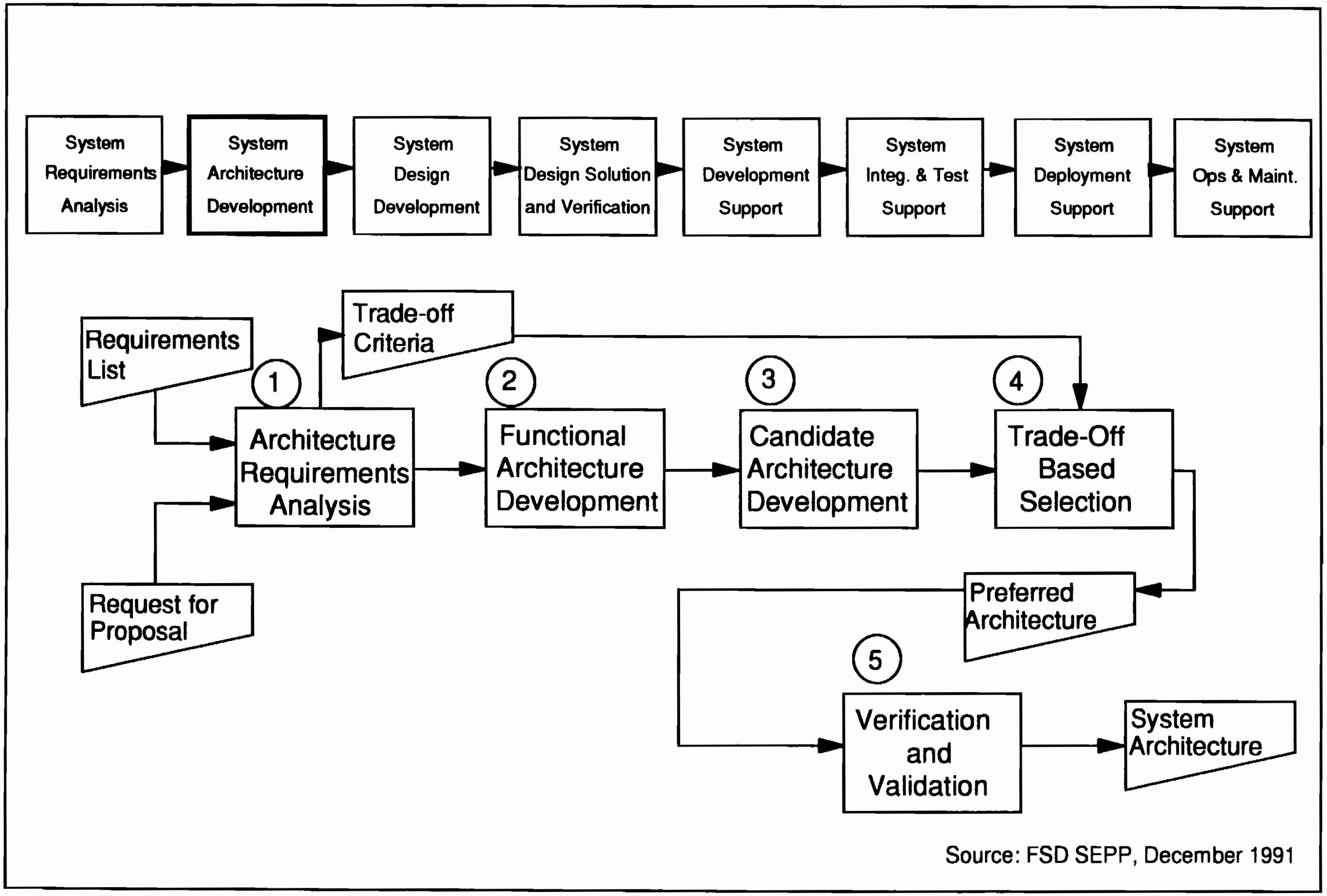


Figure 4 - Architecture Development Subprocess

The fourth task is the trade-off selection of a preferred architecture. "The preferred system architecture is that set of components and interactions which best satisfy the technical requirements and operational need within the established budget and schedule constraints."[6] The trade-off process must ensure the selection of the best effective solution from the viewpoint of the customer, the developer, and the user. A detailed explanation of the trade-off process will follow.

Task five is the verification and validation of the preferred architecture to ensure it meets functional and operational requirements. This involves ensuring testability, justifying the design, tracing each element of the design to a requirement, guaranteeing there is no overdesign or excessive performance, and providing cost versus value confidence. The goal of this task is to provide sufficient details for hardware and software selection or design and to demonstrate feasibility.

The trade-study is the heart of task four. The purpose of conducting a trade-study is to demonstrate objectivity to the customer in considering all possible solutions and identifying strengths and weaknesses of the chosen solution. In addition, it provides visibility into risks.

The first step is to create a problem statement by identifying what decision is needed. Architecture candidates must be identified using past experience and new technologies. Criteria are formulated including explicit constraints represented in the RFP as shalls, and tradeables. The criteria are weighted with respect to each other, and values are given to score each criterion. Each alternative is scored using these values and then a rating is calculated by multiplying this score with the appropriate weighting. In addition, risks and uncertainties must be assessed. The result should be the preferred alternative. A simple example of a trade-study is represented in Table 1.

Although the trade-study process appears straightforward in theory, its practice is highly subjective in that the systems engineer uses his judgement to weigh the importance of each criteria and assign a score to each alternative. There appear to be no specific tools is place today to perform cost analysis at this level. The engineer uses costs developed in previous programs for comparisons, or calls suppliers directly to get catalog prices for components under consideration. Each situation and each engineer's base of experience is unique; therefore, no standard method has been developed.

Table 1 - Sample Trade Study Matrix

TRADE STUDY NAME: EXAMPLE

PROBLEM STATEMENT:

TIME-MONITORING DEVICE FOR MILITARY AIRCRAFT

			CANDIDATES											
CRITERIA			1. DIGITAL CLOCK			2. ANALOG CLOCK			3.			4.		
CONSTRAINT			MEETS			MEETS			MEETS			MEETS		
DIMENSIONS (4"x5"x6")			(YES) NO			(YES) NO			YES NO			YES NO		
MIL SPEC PARTS			(YES) NO			(YES) NO			YES NO			YES NO		
			YES NO			YES NO			YES NO			YES NO		
TRADEABLE	W	M	VALUE	SCORE	WxS	VALUE	SCORE	WxS	VALUE	SCORE	WxS	VALUE	SCORE	WxS
1. UNIT COST ($K)	75		14	4	300	16.2	2	150						
2. SYST RELIABILITY (MTBF)	25		2846	2	50	2940	4	100						
3.														
4.														
5.														
RATING					350			250						

CHOOSE CANDIDATE 1

III. Estimating

The estimate for proposal submission is prepared during the second phase of the proposal process. A kick-off meeting is held with all principals involved in the proposal. At that time, ground rules are established, responsibilities are assigned, and a Work Breakdown Structure (WBS), that dictates the level of detail and the specific categories to which costs are allocated, is supplied.

Figure 5 shows an example of a possible proposal WBS. Although there is a military standard WBS available, the format used for a given proposal is generally determined by the program manager in accordance with customer requirements that may differ from the standard. In this example, level 1 is the contract level under which all contracted items fall. Level 2 is a break out by contract line item (CLIN). In this case, development and production are specified as distinct line items. Level 3 represents broad categories such as hardware, software, and integration and test, followed by level 4 which provides specific work packages or work breakdown structure identifications (WBSIDs).

Level: 1	2	3	4	DESCRIPTION
ALPHA0				Program Alpha Total Cost
	CLIN01			Contract Line Item 1 - Development
		HDWDEV		Hardware Development
			HWD001	Enclosure/Structure
			HWD002	Backpanel
			HWD003	Module A
			HWD004	Module B
			HWD005	Module C
			HWD006	Module D
			HWD007	Power Supply
		SFWDEV		Software Development
			SWD001	Module A Support
			SWD002	Off-line Support
			SWD003	Simulation
			SWD004	Diagnostics
			SWD005	Control Programs
		INTEG0		Integration and Test
			INT001	Module Test and Support
			INT002	Unit Test and Support
			INT003	System Integration and Test
		SYSENG		Systems Engineering
			SYS001	System Analysis and Design
			SYS002	Hardware Definition
			SYS003	Software Specifications
			SYS004	Technical Review
			SYS005	Reliability and Maintainability
		PGMMGT		Program Management
			PM0001	Configuration Management
			PM0002	Engineering Records
			PM0003	Quality Assurance
			PM0004	Program Office
			PM0005	Subcontract Management
			PM0006	Financial Program Management
	CLIN02			Contract Line Item 2 - Production
		HWPROD		Production Hardware
			HWP001	Recurring Hardware
			HWP002	Operations Support
		SUPPRT		Support
			SUP001	Program Management
			SUP002	Financial Program Management

Figure 5 - Sample Work Breakdown Structure

The traditional method employed for estimating hardware program costs has been a *bottom-up* technique. This is broken into two distinct elements: recurring hardware costs (and hands-on labor associated with its manufacture) and non-recurring costs. Figure 6 shows the essential elements of a *bottom-up* cost estimate.

Recurring Hardware Costs

An estimate of recurring hardware costs begins with drawings or verbal descriptions of a system and its specifications. This is followed by the generation of a bill of materials (BOM) within the engineering organization, generally mechanical engineering. The BOM is a representation of all the parts required to build each assembly in the total product. The next step is the tedious process of applying an estimated cost to each part in the extended parts list. Hands-on labor and procurement of the parts are considered in this step. Labor charges may be broken into assembly, quality control, and test operations according to a labor routing. Procurement may come in the form of general purchases, major subcontract purchases, and operational subcontracts.

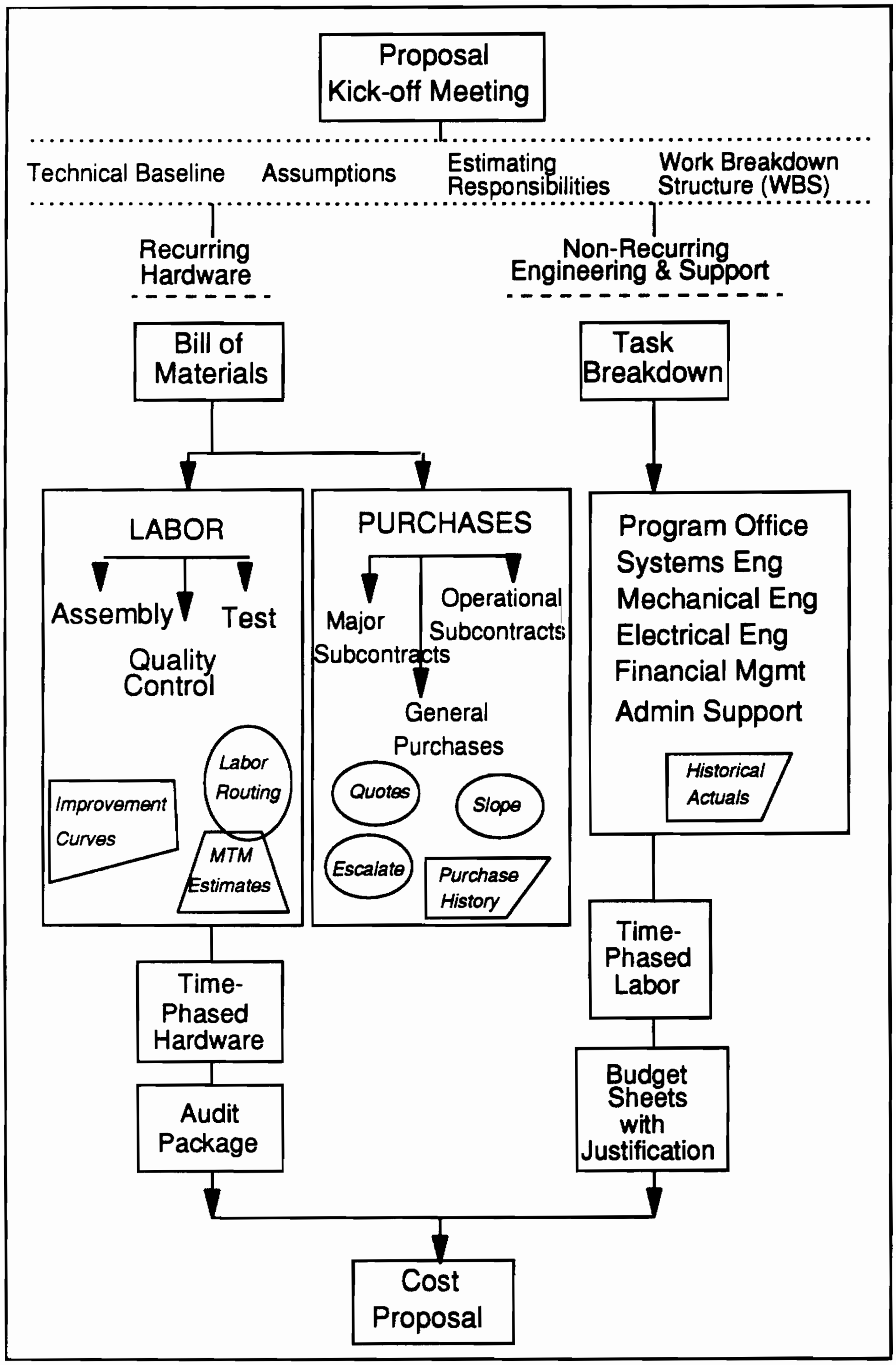


Figure 6 - Bottom-up Estimating Flow (Cost Proposal)

Operational subcontracts refers to purchases of labor from an outside vendor. Necessary parts and materials are supplied and the subcontractor assembles, tests, or performs the required tasks. Major subcontract purchases are distinguished from general purchases when total procurement from a particular vendor exceeds a minimum dollar value for that contract (for example, $1M).

Estimates for the procurement of parts may require the use of purchase history from past contracts or quotes from potential suppliers. Purchase history is a database that lists all past purchase orders placed for each part. If a particular part has never been procured, a part that is *similar-to* the part in question in terms of form, fit, and function is used for the estimate. Purchase history is *sloped* to take into account price breaks for volume purchases and *escalated* to account for differences in the time between the history and the current order. For large dollar items, quotes from various qualified vendors are solicited and the most cost-competitive price is used for the contract. This process of receiving official quotes from subcontractors generally takes between 8 and 12 weeks.

In addition, a number of adders including procurement burden, scrap, and obsolete must be tracked through history and applied as appropriate. Procurement burden is the cost to administer the purchase order process. A percentage is generally applied across all programs. Scrap are those parts that fail during manufacturing and can no longer perform their required function, and obsolete are those parts that were procured for a contract but no longer fit into the current design, or were left over from a volume purchase that exceeded requirements. It is difficult to accurately predict the amount of scrap and obsolete a project will generate; previous programs are used for a basis of estimate.

In order to estimate the labor to complete a task for the current proposal, actual costs must be tracked in similar programs and new information including manufacturing and test plans must be incorporated. In addition, improvement curves related to learning and experience must be considered. The theory behind improvement curves states that the time required to do a job will decrease each time that job is repeated, and the amount of the decrease will be less with each successive unit up to some ultimate performance. Improvement curves are expressed in terms of a slope (for example, 80%) that is based on the contractor's past performance. If actual historical data is not available, motion time measurement (MTM) estimates may be required. MTM is a standard of industrial engineering where individual movements to perform each task are timed and accumulated. This is a time-consuming process that is used only as a last resort.

It is generally thought that 80% of the cost of hardware is found in 20% of the parts. But in a *bottom-up* effort, equal time is often given to estimating costs for the remaining 80% of the parts that go to the lowest level, including nuts, bolts, and screws. Another approach is to estimate the remaining low-value parts as a function of the high-value parts.

Although there are a number of semi-automated systems available to aid the cost estimator, the current process is very labor intensive and there is a great deal of opportunity for error. The possibility of a high-value part or group of parts being left off the BOM and therefore not included in the cost estimate is a potential disaster. The use of manual entries also introduces the risk of typographical errors.

The result of these efforts is a time-phased estimate of recurring hardware costs in the form of procurement dollars and labor hours, to which current labor and burden rates are applied. There are many variables to consider and each detail is open to scrutiny by the auditing agency. However, this type of estimate is considered the most *supportable*. Within FSC Manassas, the cost engineering organization has primary responsibility for estimating recurring hardware costs for a proposal and supplying supporting data for these estimates. All of the details needed to support the estimate must be supplied (in an easy to understand format) in an audit package (included in the cost proposal).

Non-Recurring Costs

The non-recurring effort (NRE) includes all engineering activities, program management, and support functions. These costs are generally estimated by the performing organizations. In many cases, a heuristic approach is taken to estimate these costs. For instance, an experienced engineer may relate this task to tasks performed in a similar, previous program and make a judgement as to its relative difficulty. Actual labor charges for the previous program are then adjusted accordingly.

Another approach would be to list each task to be performed and each work product to be delivered and apply a discrete estimate for each element. Each task to be performed must be considered and estimated individually in this *bottom-up* approach. In addition, several organizations may contribute to a single task. For instance, in the WBS shown in Figure 5, the work package for development of Module A may include effort for electrical engineering, mechanical engineering, technology development, such as microcircuits, as well as recurring hardware. Depending on the complexity of the project and the level of detail, the number of estimates required may number into the hundreds.

Labor estimates are time-phased across the life of the project in order to apply appropriate labor and burden rates. Budget sheets, as illustrated in Figure 7, are used to represent the estimated labor to complete each task, as well as computer time, purchases, and so forth. Using this method, the estimator has control and ownership of his or her estimate.

IBM	PROPOSAL MIC BUDGET SHEET – FORM C
PREPARED BY: JOE SMITH	EXT: 1234 DATE: 4/1/90 Page 1 of 1

1 CD	2-16 WBSID / PN	17-22 ORDER	24-26 DEPT	27 SHIFT	31-35 QTY	WBS DESCRIPTION
3	SYS001		ABC	1	–	SYSTEM ANALYSIS AND DESIGN

LGC 28-30	RE-VISION 36-38	DO NOT FACTOR 39	OVER TIME 40	CREDIT 41	HOURS 42	NO PROC BURDEN 43	PERCENT 44	REPLACE 45	DEPT WORKING TITLE
VARIOUS		IIIII			X		IIIII		SYSTEMS ENGINEERING

TIME PERIOD 44-49	MANPOWER (MONTHS IN TENTHS) A 50-54	B 58-62	C 66-70	D 74-78	E 82-86	NONENG 90-94	COMPUTE 98-105	TRAVEL 109-116	PURCHASE 120-127	SUB-CONTRACT 131-138	IPT / OTHER NVA 142-149	EOC 153	OTHER VA 154-161	CSP 165-174
9301	80	80												
9302	80	80												
9303	80	80												
TOTAL	240	240	0	0	0	0	0	0	0	0	0	0	0	0

WORK DESCRIPTION:
ANALYZE SYSTEM REQUIREMENTS AND DEVELOP SPECIFICATIONS. DEVELOP SYSTEM ARCHITECTURE AND COORDINATE HARDWARE AND SOFTWARE DESIGN.

I certify that this is my best estimate to perform the work described on this worksheet.

ROBERT JONES — WORK PACKAGE MANAGER NAME (Please print)
2345 — EXT
Robert Jones — SIGNATURE

Figure 7 - Sample Budget Sheet

IV. Control

Once a contract has been awarded and performance begins, a system must be put in place to control technical aspects, cost, and schedule. A technical management plan (TMP) is used by program management and systems engineering to facilitate this. Risk management is a major aspect of a sound technical management program.

Technical Management Plan

The TMP, as it is referred to by FSC, is very similar to the Systems Engineering Management Plan (SEMP) described by Blanchard in *Systems Engineering and Analysis*, and is illustrated in Figure 8.[7] The purpose of the TMP is to describe how the contractor intends to plan, conduct, and control the effort needed to accomplish assigned technical tasks. In FSC, it is comprised of four parts. Part 1 describes the technical program planning and control. Part 2 describes the systems engineering process employed to implement Part 1. Part 3 describes the application and participation of the engineering specialties. An additional section, Part 4, is included to reflect FSC's method of providing status to the customer for those points considered in Part 1.[8]

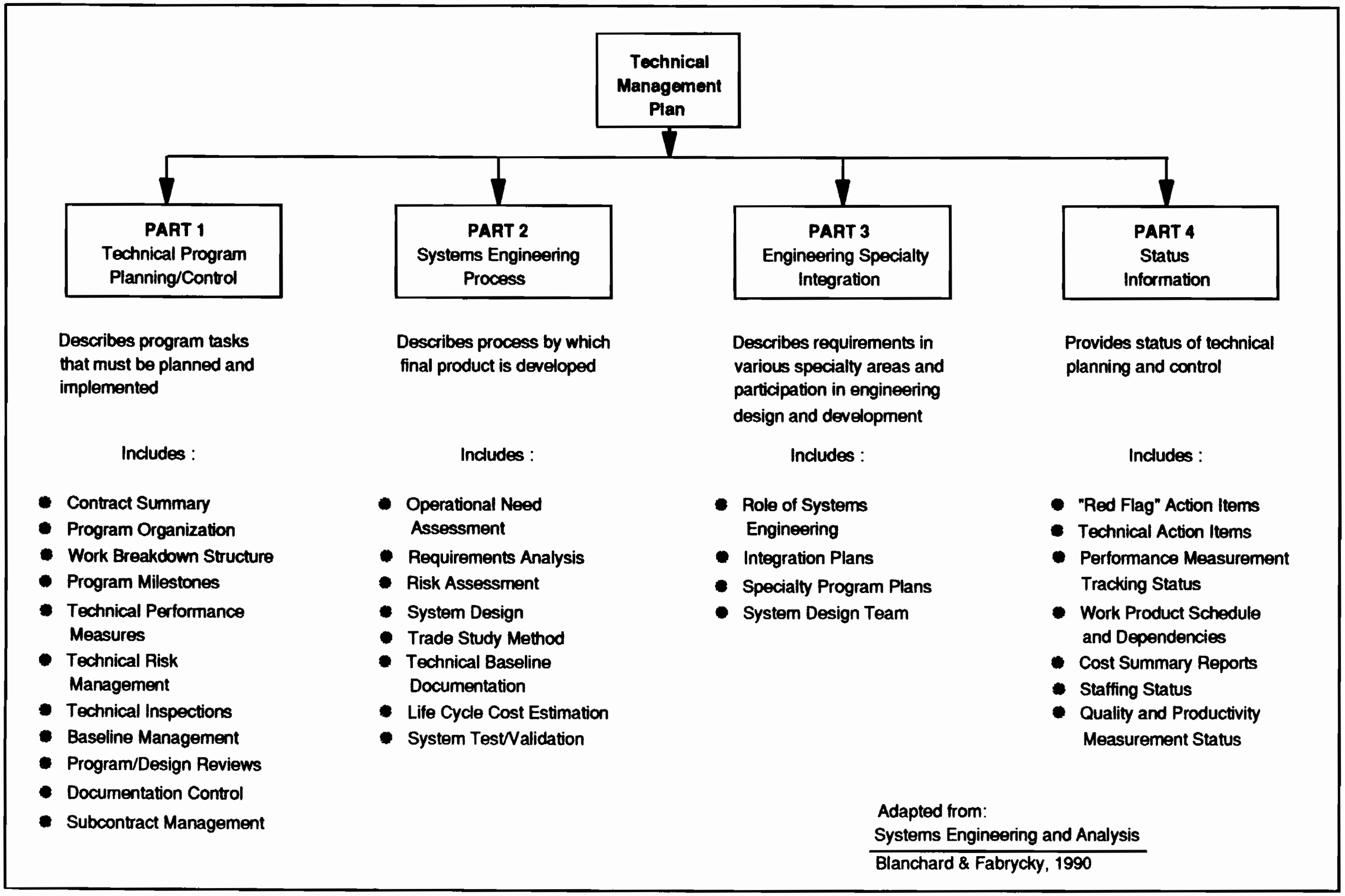


Figure 8 - Technical Management Plan

An overview is presented in Part 1 to state the system needs and describe the system in terms of an operational, functional, and physical view. It also summarizes the contract customer, SOW, and key assumptions made. The planning section of Part 1 explains the program organization and WBS, technical program integration, and program milestones. The control section focuses on technical risk management and performance measurement and inspections, by describing what processes will be used to identify, assess, quantify, and contain technical risk, determine progress, and ensure required technical work completion. In addition, other control mechanisms such as baseline management, documentation control, program and design reviews, and subcontract technical management are discussed.

Part 2 of the TMP is designed to show the specific processes and procedures used to integrate the technical activities needed to create an effective system and relate them to the technical planning and control methods described in Part 1. This includes an overview of key systems engineering concepts and management processes. It also describes the processes for technical risk management, incremental development, baseline documentation, life cycle cost (LCC) estimation, system test and validation, and systems engineering quality and productivity measurements.

Part 3 explains the role of engineering specialties in system design and how they are identified and integrated. It also explains the role of systems engineering in engineering specialty work products, such as ensuring consistency with system requirements, baseline design, test plans, and schedule and cost constraints.

Part 4 is used to report and track project technical status. It is normally published separately from Parts 1-3 and contains current status information in the form of schedules, action items, and dependencies used to control the systems engineering process.

Risk Management

Risk management is an important concept in the overall management process. Risk management includes several related actions: risk assessment, risk analysis, and risk handling. Risk assessment is the process of examining a situation and identifying the areas of potential risk. The purpose of risk analysis is to discover the cause, effects, and magnitude of the perceived risks and to develop and examine alternative options. Risk handling includes techniques and methods developed to reduce or control risks.[9]

Various methods for identifying risks are available. Models and life cycle cost analysis are two of the recognized sources for identifying risks as well as describing and quantifying the magnitude of that risk. Risk can be simply modeled as the interaction of two variables, probability of failure, and the effect or consequence of that failure in terms of technical performance, cost, or schedule.

"A technical risk assessment system should provide all levels of management with a disciplined system for early identification of technical uncertainties, a tool of instantaneous assessment of current program status, and early key indicators of potential success or failure."[10] Risk management in terms of cost is examined in detail here. According to DoD documentation, five important risk areas have been identified: design, test, production, cost, and management. The suggested key indicators for each area are shown in Figure 9. Cost issues (marked by asterisks) are found in three of these major areas.

Design to cost (DTC) is one risk indicator in the design area. According to DoD directive 4245.3 (the principal policy statement on cost), cost "is a parameter equal in importance to technical and supportability requirements and schedules."[11] DTC is a method of establishing cost goals for the design phase and tracking progress toward those goals.

DESIGN	TEST	PRODUCTION	COST	MANAGEMENT
■ Weight	■ Critical Test Network	■ Transition Plan	* Cost Performance Index (CPI)	■ Major Technical Risk
■ Length	■ All Test Reliability Growth	■ Delinquent Requisitions	■ Schedule Performance Index	■ Technical Specification Verification
■ Endurance		■ Incoming Material Yields	* Estimate at Completion (EAC)	
■ Center of Gravity		■ Manufacturing Yields	■ Management Reserve	
■ Throughput		* Unit Production Cost		
■ Memory Utilization				
■ Drawing Releases				
* Design to Cost (DTC)				
■ Design Maturity				
■ Engineering Man Hours				
■ Engineering Changes				
■ Failure Activity				

Source: Technical Risk Assessment
Department of Navy, 1986

Figure 9 - Key Program Risk Indicators

During performance of a contract, control of costs centers around the contract WBS. The total negotiated contract value is broken down to the level required by the customer. In most cases, this WBS will be different from that used during proposal preparation. With the budget in place, costs are accumulated against each WBSID and compared on a routine basis to the budget. The ratio of budgeted cost of work performed (BCWP) to actual cost of work performed (ACWP) is termed the cost performance index (CPI) and is one of the key indicators for cost performance.

The CPI ratio is compared to some preestablished standard. Based on the actual costs accrued, an estimate at completion (EAC) is developed and compared to the budget. EAC is also a key indicator and can be developed in several ways. One method involves estimating *bottom-up* all the remaining activities. Another assumes that the remaining costs will not change so that the expected overrun will be exactly at the current value. A third and possibly more realistic method is an estimate of the remainder of the program based on the CPI. The difference between the EAC and budget is the cost variance. Very complex computerized systems exist to track this progress, but variances must still be explained outside the automated system.

V. Proposed Solution

It has long been recognized that estimating program costs should be an iterative process. Estimates of the bid price are required at key milestones during the proposal phase with each estimate reflecting a greater degree of accuracy than the preceding estimate. Program cost estimates are driven by and should be used to drive the technical solution. However, there has never been an efficient tool to carry this out systematically. Parametric costs models offer a solution to this problem.

Parametric Models

A parametric cost model is a representation of the traditional methods used to derive cost estimates. Models are being widely used in the arenas of hardware and life cycle costs, software, management, and risk analysis. Parametric methods are based on cost estimating relationships (CERs), or relationships between cost, schedule, and measurable attributes of a system. These CERs are built into simulation models in the form of statistically and logically supported mathematical equations that relate input variables or parameters to cost. An example of a CER would be *cost per square foot* for a building. The equations and CERs are based on history from similar projects. In addition to cost, models can be used to project schedules, reliability factors (for instance, mean time between failures), productivity factors (for instance, lines of code per labor month), and learning. A simplified flow of a parametric estimating process is illustrated in Figure 10.

Non-cost variables such as physical, personnel, and project attributes of the system are input to the model. The model uses internal equations (based on CERs), technology and improvement curves, and tables based on historical data, and provides cost outputs. Historical data and costs are also provided to the model to reflect individual performance of the estimating company.

BOMs and labor routings (used in a traditional cost estimate) are not used in a parametric cost estimate. Although this may represent a substantial benefit, BOMs and routings are also not part of the parametric output. This is a recognized limitation to models as cost estimating tools in an environment where government customers are accustomed to receiving this detail.

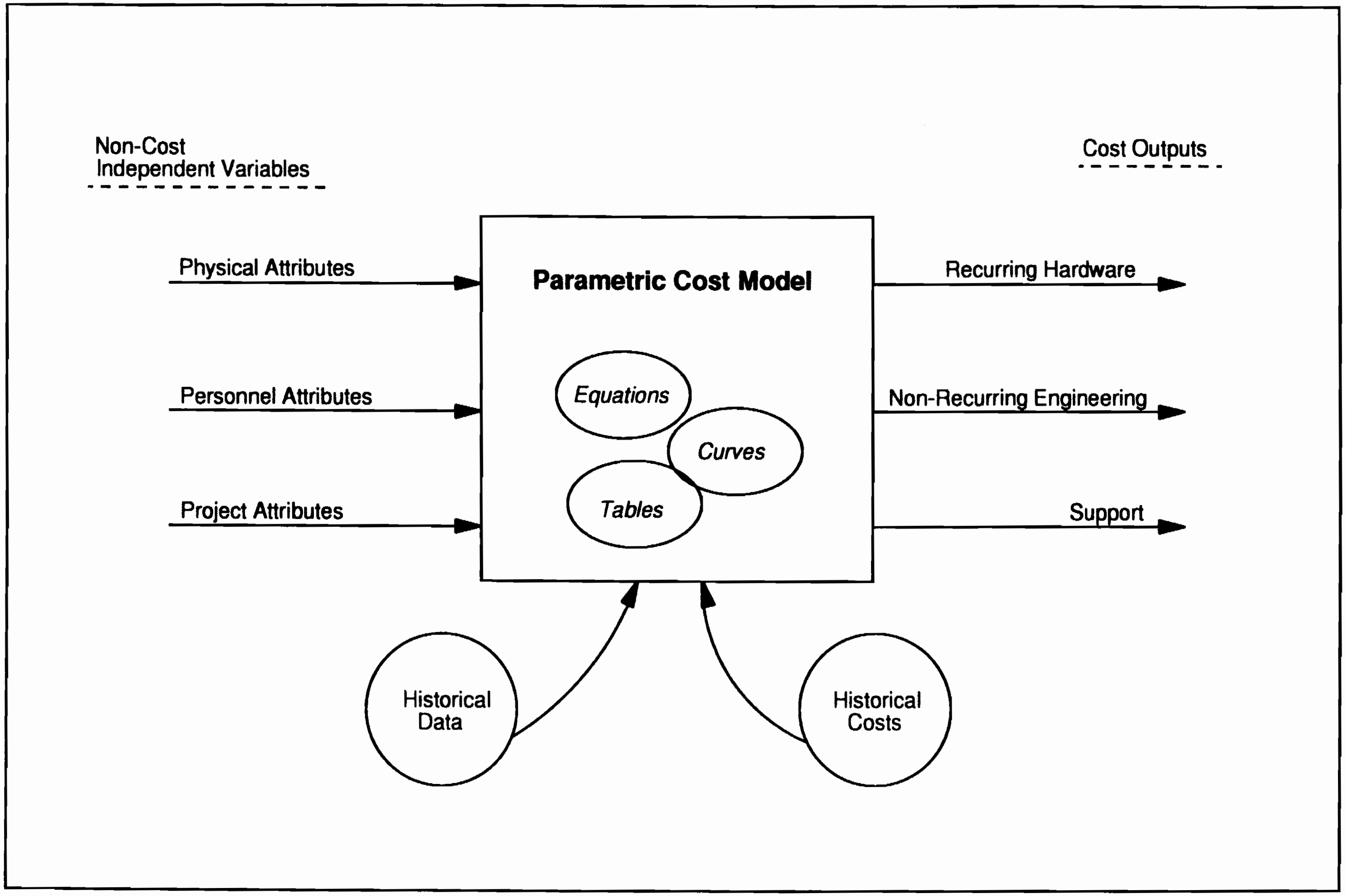


Figure 10 - Parametric Estimating Flow

Other problems exist with the use of *canned* commercial cost models. The data behind the CERs are unknown to the user and cannot be verified via statistical methods to be applicable to the program that the user is estimating. The specifications of the variables and estimating procedures and assumptions inherent in the model are also unknown. However, the validity and applicability of the CERs to an organization or a product can be assessed. The results of the models can also be evaluated in terms of previous estimates. A parametric cost model should not be viewed as a black box that contains answers. Instead, it is an aid to an estimator and its accuracy is dependent on the skill of the user.

There is an interrelationship between the planning, estimating, and control aspects of any given project. Parametric modeling is a tool that can be used effectively in all three areas and can help tie the three together. The planning aspect involves the conceptualization and development of a solution to a customer problem. Parametric modeling puts the cost estimating process at the front-end where design decisions are being made. Trade-off studies between design alternatives can be performed quickly and cost-effectively using parametric analysis. Parametric cost models can also be used in the early planning stages to support a bid/no-bid decision before entering a procurement competition.

Cost estimating takes place during the proposal process after an architecture has been chosen. Parametric cost models can be used to project system cost and schedule, including hardware, software, and engineering effort. This method requires a fraction of the number of people involved in a *bottom-up* effort working for significantly less time, and because fewer people are involved, there is less chance for miscommunications. The estimate of recurring hardware costs and non-recurring engineering effort are integrated, so assumptions and ground rules are assured to be consistent.

The management or control aspect of a program becomes important once a contract has been won and performance must meet the specifications and cost agreed upon during negotiations. Models being used as measurement tools can make an important contribution to Market Driven Quality (MDQ) in risk analysis and cost and schedule control. Data obtained from this phase of a project is also used to calibrate the models for future proposals.

The feedback loops between these three areas, planning, estimating, and control, are illustrated in Figure 11. The design established during the planning stage becomes the basis of the estimate prepared for proposal submission. Conversely, the estimate of acquisition cost and LCC can be used to influence the architecture chosen during trade-studies. The estimate made during a winning proposal and represented in a WBS, becomes the focus for measuring performance. The estimate must be based on calibrations obtained through the performance phase of previous, similar programs. This history is also used in the form of project experience to create new solutions in future planning activities.

PRICE Model

For purposes of illustration in this text, the PRICE hardware model is used. This model is a computerized method for deriving cost estimates of electronic and mechanical hardware assemblies and systems, especially in the aerospace industry. PRICE H was developed by RCA Corporation for its own use in the 1960's and was made commercially available in 1975. Today, it is part of a family of models owned and operated by General Electric Price Systems in Moorestown, New Jersey.

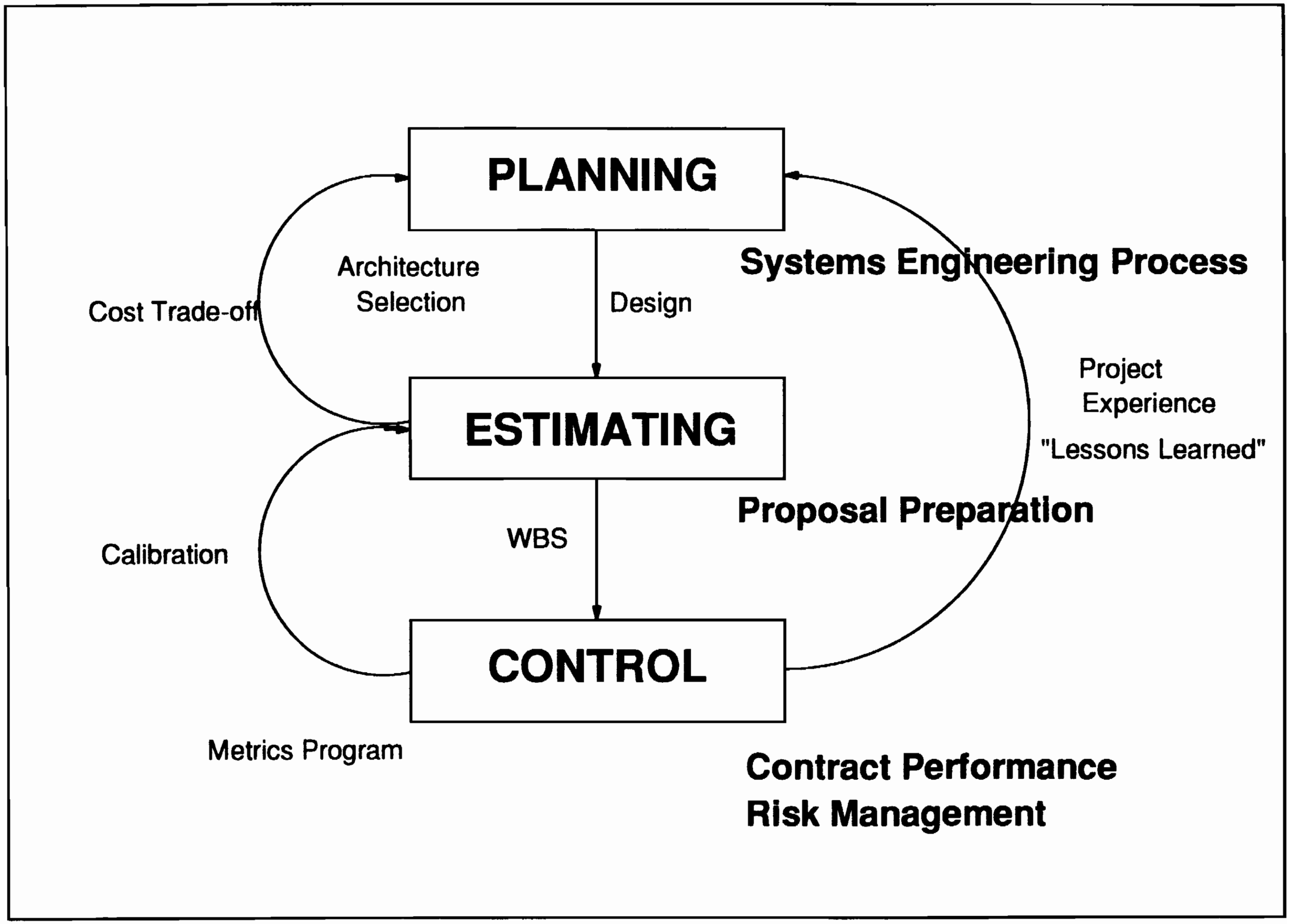


Figure 11 - Interrelationship of Three Project Areas

PRICE H has a broad range of applications and it is the predominant hardware parametric cost model used throughout U.S. industry and Federal agencies, as well as international companies.

The fundamental data used by the PRICE H model includes quantities, schedules, geometry (for example, size and weight), amount of new design effort, operational environment, and manufacturing complexities of structure and electronics, among others.

Although PRICE H was developed for the aerospace industry, it can be customized to other environments and changing technologies. It provides flexibility to describe products and processes that are beyond the experience upon which the model is based. For instance, FSC cost engineering has used PRICE H to estimate commercial off-the-shelf hardware, optics, and x-ray technology, in addition to military hardware.

The PRICE H user interface provides a structured approach to estimating that helps avoid oversights and errors. Input parameters that are omitted or are beyond the range of acceptability for the model are identified to the user.

Another advantage of the model is its capability of calibration. Calibration is a process by which the model outputs are tailored to a specific organization or product. Through calibration of the model, previous experience and specific characteristics of the using organization are incorporated. Even though the estimating procedures and relationships cannot be controlled by the user, the estimate is a product of historical data provided by the estimator.

Use of this particular model should not be viewed as an endorsement of its superiority to other available models. There are other hardware models available that will not be addressed in detail. The System Evaluation and Estimation of Resources for Hardware (SEER-H) model, created by Galorath Associates is an example. SEER-H is relatively new on the market and has not been widely tested as yet. For this reason, it is not being used as the case study for this text but serves as evidence that the issues and concerns related to PRICE are being addressed in new development activities. The usefulness and applicability of any particular model must be determined by the user based on individual requirements. Several criteria that should be considered are discussed in section VI.

Limitations of the Model

In addition to the general concerns about parametric cost estimating models, there are several limitations specific to PRICE H. The output of the PRICE H model is divided into seven categories: drafting, design, systems engineering, project management, data, hardware, and tooling and test equipment. The model cannot break out the recurring cost between elements such as procurement, or purchases, and hands-on labor, nor can it break out NRE by task. If the estimate is required at a greater level of detail than that provided by the model, this can be accomplished by relating the cost outputs to previous estimates and actual cost data.

The PRICE H model is weight-based, meaning that the weight of the subassembly or system is a major cost driver. This is seen as a problem by many who interpret this as *pricing technology by the pound*. In all fairness, the model makes a distinction between structural and electrical parts and further breaks the electronics into types of technology. Still, this issue persists. New models such as SEER-H recognize this drawback and therefore do not include weight as a major cost driver, instead relying on the complexity of technology to estimate costs.

Another problem of the PRICE H model is security. The model operates on a dedicated mainframe computer at General Electric in Moorestown and users access it via modem. Many managers, especially those in charge of programs in direct competition with G.E., view this as a risk to confidentiality of data. There is no method to prevent access to phone lines from outside interests. However, PRICE Systems (the General Electric division that provides support and consultancy for the models) maintains that information obtained in this fashion would not be meaningful. In addition, the PRICE Systems group is an independent arm of General Electric Company that does not share information with other divisions within G.E. Other models, that are based on a personal computer (and are therefore self-contained on the estimator's desk), will eliminate this issue.

Use of Parametric Models in Planning

The planning phase is the first area that is discussed in terms of using parametric models. After a business opportunity has been identified and evaluated, a decision must be made as to whether or not the contractor will pursue it. Using an initial technical baseline developed by the systems engineers, parametric models can be used to support a bid/no-bid decision in terms of cost. In other words, the model can provide an estimate of cost that can be compared to the cost bogey (or target) generated through competitive analysis. Once a decision has been made to bid on the contract, it is at this point, during conceptualization, where parametric modeling can be effective in helping to determine the optimum solution based on cost, schedule and some performance characteristics. Using a model to assist in this process introduces a level of objectivity and gives consistent, repeatable results even where alternatives are dissimilar.

As an example, consider a customer objective of monitoring time of day in a military aircraft. (Data for all examples in this text are fabricated for the purpose of illustration.) The time-keeping unit shall occupy a predetermined place in the system and has a *footprint* of 4 inches high by 5 inches wide by 6 inches deep. Based on this information plus quantities and schedules, the systems engineers develop two alternatives, a digital clock and an analog clock.

We will circumvent the details and assume that the systems engineers follow the prescribed steps to determine these candidate architectures. It is during the fourth step of the systems engineering process (trade-off selection of the preferred alternative), where parametric models can be used. The trade-study is based on many criteria including constraints and tradeables. In this example, the *footprint* is a constraint and was identified as such in the RFP. The schedule and quantities of units to be delivered were also specified.

PRICE H can be used to estimate the acquisition cost of these two alternatives at the conceptual level. There is no need for mechanical engineering to generate BOMs in order to estimate the cost of each individual part, and there is no need to involve the procurement organization who would normally be responsible for obtaining quotes from possible vendors. Instead, parametric inputs are used to describe the hardware unit. Table 2 summarizes the inputs required to perform the cost trade-off.

Quantity, geometry, schedule, and operational environment are the same for both alternatives; they are specified as constraints. The alternatives differ in terms of design effort and complexity of the equipment. The engineering complexity value is based on the experience of the design team and scope of the design effort. In both cases, the design effort is based on a new design that is within the established product line of the company. However, the design team has more experience with digital technology than with analog technology. Included in the PRICE manual are tables that quantify such qualitative parameters as complexity or experience. The definitions of *normal* versus *extensive* or *mixed* experience are open to interpretation. The value of 1.0 for Alternative A is taken from the PRICE table and indicates that the engineers have *normal* experience in completing similar type designs. The table value of 1.1 for Alternative B means that the engineers have *mixed* experience; some are familiar with this type of design and others are new to the job.

Table 2 - Parametric Inputs for Trade Study

INPUT VARIABLE		ALTERNATIVE	
		A - Digital	B - Analog
Quantity			
	Number of Prototypes	5	5
Geometry			
	Estimated Total Weight	6 lbs.	6 lbs.
	Weight of Structure	4 lbs.	4 lbs.
	Volume	.07 cu.ft.	.07 cu.ft.
Schedule			
	Development Start Date	January 1993	January 1993
Design Effort			
	Percent of New Electronics Design	90 %	100 %
	Percent of Repeated Electronics	20 %	0 %
	Percent of New Structure Design	100 %	100 %
	Percent of Repeated Structure	50 %	50 %
	Engineering Complexity *	1.0	1.1
Operational Environment			
	Platform *	1.8	1.8
Manufacturing Complexities			
	Complexity of Structure *	5.78	5.78
	Complexity of Electronics *	7.94	8.09

* refers to table value found in PRICE manual

For both alternatives, a new structure must be developed and there is 50% redundancy to the design. This means that the engineers design 50% of the structure and the remaining 50% is a duplicate of that design. For the electronics, all of the analog solution needs to be designed while 10% of the digital solution is taken from previous programs. There is no redundancy in the analog design, but 20% of the digital electronics is repeated.

The manufacturing complexity values indicate the difficulty of producing the technology. The table value for an aluminum machined casting is used for the structural complexity. The electronic complexities taken from the table are based on the assumption that the majority of components will be integrated circuits. This assumption is adequate for the purpose of an estimate at this level. Once an alternative is chosen and its cost is estimated for the proposal, more detail will be needed to improve the precision of the estimate. This will be illustrated in later sections.

The purpose of this example is to show that the type of information needed to make a cost estimate at the conceptual level is readily available from the RFP and the systems engineers' experience. The time required to make such an estimate is minimal; if changes to the specifications are made, revisions to the estimate can be made quickly and efficiently. There is the added advantage of using the same methodology to estimate the cost of both alternatives.

The results of the PRICE model run are summarized in Table 3. Although the recurring hardware costs are not significantly different, the development costs of Alternative A are appreciably less than the costs for Alternative B. This seems reasonable based on the assumption that the design team has more experience with digital technology and less new design is required. There is also a two-month difference in development time between the two solutions. There is a slight improvement in the reliability expected with the analog technology, but this does not appear to justify a change to the more expensive design.

Table 3 - Parametric Outputs from Trade Study

OUTPUT CATEGORY	ALTERNATIVE	
	A - Digital	B - Analog
COST ($K)		
Non-Recurring Engineering (NRE)		
Drafting	102	161
Design	341	560
Systems	47	86
Project Management	42	74
Data	15	28
Recurring Hardware		
Prototype	70	81
Tooling & Test Equipment	9	10
Total Project Cost	626	1,000
SCHEDULE		
Development Cycle (months)	13	15
RELIABILITY		
Mean Time Between Failures (MTBF)	2846	2940

NOTE: A definition of each cost category is provided in Appendix A, taken from the *PRICE H User's Manual.*

To expand on this example, it was determined during competitive analysis that the customer was probably willing to spend no more than $500,000 for the contract. This would have been established as the cost bogey. The results of the initial PRICE runs could be used to show management that there was some risk involved in competing for this procurement, and in such a way, influence the bid/no-bid decision.

Of course, the clock in this simple example is a single element of a larger system. Taking a system view might entail estimating many hardware units and possibly a software component and the integration of all these elements. This could also be accomplished using parametric models. While the individual hardware units and their integration could be modeled using PRICE-H, the software component would require a unique model; several are available. Software models base cost on an estimation of source lines of code (SLOC) to be delivered, productivity of the performing organization, and personnel and program attributes, among others. The outputs from hardware and software models would be used to establish a total system cost.

In addition to acquisition cost, LCC has become an increasingly important factor to be considered. LCC is the total cost to the government for acquisition, ownership, and disposal of a system over its entire life. Historically, a low initial acquisition cost for hardware has not assured a low LCC. In fact, the opposite is often true. Efforts to minimize LCC are most effective in the conceptual and early design stages when alternatives are being identified and selected. In this example, if the case had been that LCC was designated a criteria instead of, or in addition to acquisition cost, the PRICE-HL model could be used. This is a hardware life cycle cost model that builds on the output from PRICE H using some additional parameters about deployment theaters, maintenance concepts, and spares.

Use of Parametric Models in Estimating

Proposal preparation and submission is the next area where parametric modeling can be incorporated. After an architecture has been chosen, the cost estimator will expand on the estimates performed for conceptual level analysis. A greater level of detail and historical data from past programs are required. Models can be employed to develop costs for the proposal, or if parametric methods are not appropriate, they can be used to substantiate costs derived by *bottom-up* techniques. In addition, they can be used to aid in source selection by establishing vendor *should-costs*.

One problem that should be anticipated in the application of any new approach is the resistance to change, specifically the resistance of decision-makers. Once program managers and executives of the contractor are convinced that a new method will make an improvement to the existing estimating process, the task of convincing the customer is paramount.

Changes to a contractor's estimating system are subject to rules imposed by Federal Acquisition Regulation (FAR). According to FAR 215:811, "'Estimating System' is a term used to describe a contractor's policies, procedures and practices for generating cost estimates which forecast costs based on information that is available at the time." In addition to information about a contractor's organizational structure and internal controls, it includes the methods and techniques used for estimating, the process for accumulating historical costs, and the analyses used by a contractor to generate cost estimates and other data included in proposals.[12]

To be considered adequate, a contractor's estimating system must provide for the use of historical experience where appropriate, integrate information available from other management systems, and provide for internal review of and accountability for the adequacy of the estimating system. This includes the comparison of projected results to actual results and an analysis of any differences. Cost and performance data must be kept accurate, complete, and current and it must be available for use by the estimators.[13]

The Defense Contract Auditing Agency (DCAA) is responsible for auditing the proposals of all competitors. "DCAA has long viewed parametrics as an acceptable estimating technique," based on a set of specified criteria. These criteria are: logical relationships between cost and non-cost independent variables, verifiable data, significant statistical relationships, reasonable accurate predictions, and proper system monitoring.[14] This policy was disseminated to site auditors via a memorandum and it is these individuals who must audit and approve the estimating methods used.

It is the contractor's responsibility to show that their use of parametric estimating meets these criteria and that their estimating system is adequate. In order to do this, methods used for cost collection and structures developed for bidding proposals must be made consistent. The estimating system must also be based on the contractor's historical performance. Further discussion about the collection and maintenance of cost data will be found in the following section. An estimator must recognize that the output from the parametric model may or may not be representative of the effort required by their company. However, some relationship exists. Historical data from past, similar programs is used to determine this relationship and calibrate the model. Calibration is a method to tailor the model to a company's past performance in terms of product and organizational attributes.

According to DCAA, "as with the use of any estimating relationship derived from prior history, it is essential in the use of parametric cost estimating relationships for the contractor to document that work being estimated is comparable to the prior work from which the parametric data base was developed."[15] When using PRICE, the platform value or operating environment is important in this respect. It would not be valid to compare programs that were dissimilar due to special constraints put on each of the platform categories. For instance, military flight applications require different considerations than commercial applications.

Revisiting the example presented in the previous section, consider that the system for which the digital clock was being developed is part of a proposal effort. As the hardware design matures, additional detail becomes available about the components that will be used. This information is incorporated into the original PRICE estimate by adjusting the complexity factors. For instance, if the system developer specifies that 20% of the design will be display equipment with digital equipment comprising the remaining 80%, a new manufacturing complexity of electronics (MCPLXE) would be generated to reflect this. If the original aluminum structure design is being enhanced with steel reinforcements, the manufacturing complexity of structure (MCPLXS) will be modified, identifying a more difficult manufacturing process.

In addition to these changes, the model must be calibrated to past performance by analyzing a similar program that is already completed. For this example, the development and production of a speed monitoring system for a military aircraft is chosen. Assuming that documentation showing similarities between the two programs is present and substantial, the attributes and costs for the completed program can be used as a calibration point. A flow of the calibration process is depicted in Figure 12.

Calibration of the PRICE H model begins with analyzing the completed program in a similar manner to that demonstrated earlier. In other words, input variables that describe program and personnel attributes are collected. The model is run using these inputs, and the cost outputs from the model are compared to the actual costs accrued for the given program.

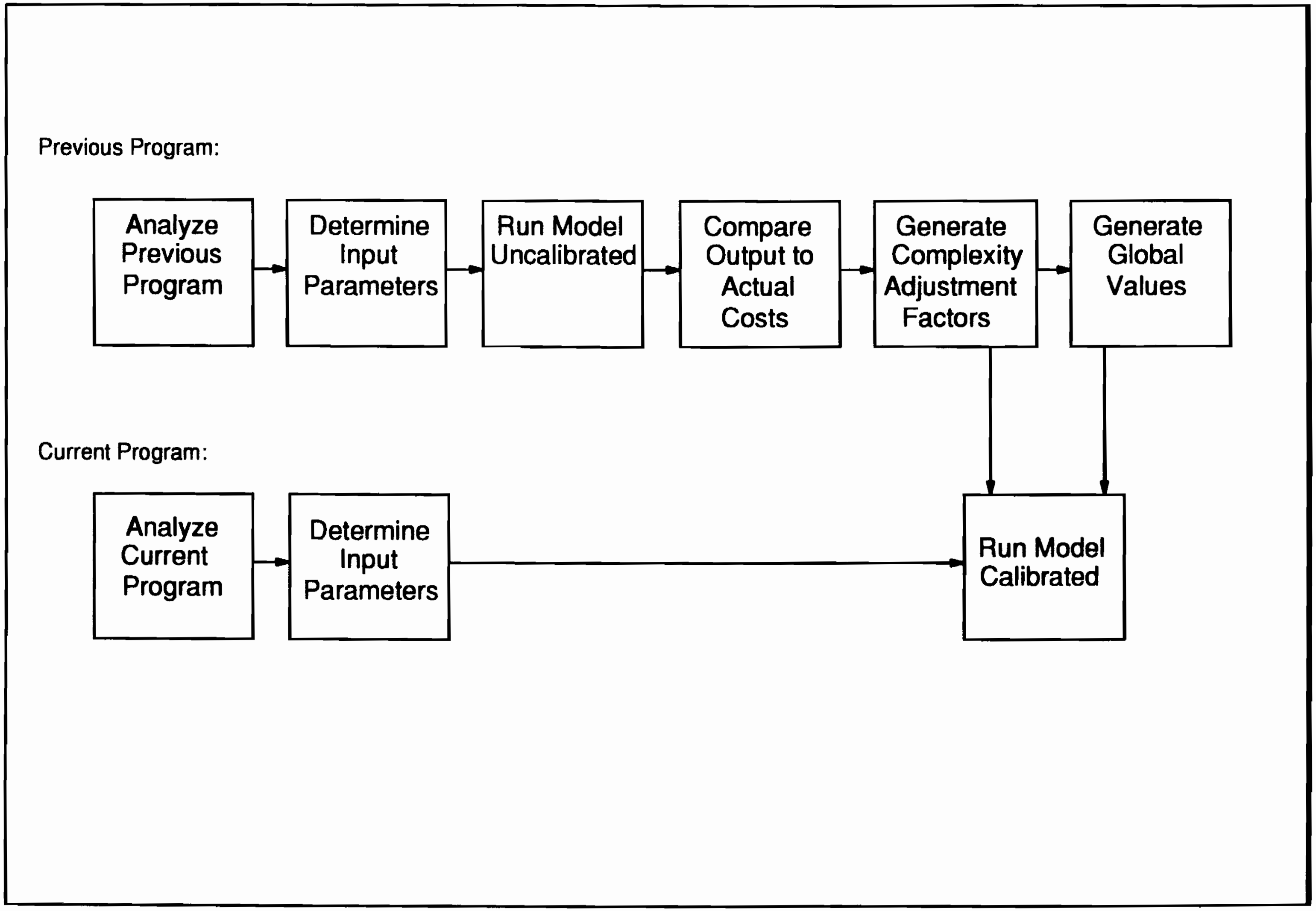


Figure 12 - PRICE Model Calibration Flow

Product calibration involves adjusting the complexity values (MCPLXE and MCPLXS) so that the recurring hardware cost output from the model equals the actual cost incurred. Using these new complexity values, another run of the model is completed to give an estimate of the non-recurring costs. Again, these costs are compared to the actual costs and *global* values are used to adjust the cost outputs. These *global* values are linear multipliers of the non-recurring cost outputs that reflect individual organization characteristics.

For instance, according to PRICE H, systems engineering is the effort needed to transform system requirements into specifications. The definition according to FSC involves much more, as discussed in previous sections. *Global* values are used to bridge this difference. Once complexity adjustment factors and global values are established from the previous program, they are applied to the PRICE estimate for the current program. Calibration is also the only way to ensure that cost estimates derived using models are reasonable and accurate predictions of the contractor's performance.

Although there is a general definition for the content of each PRICE output category, this is a high level cursory view and no further visibility into the individual tasks that comprise each cost category is available. Calibration can also be used to estimate the breakout of these costs according to some previous program that has been determined to be similar. For instance, if it was found that 50% of the support costs estimated in Program A were expended by the program office, that percentage can be used to determine the program office content of the support costs estimated for Program B.

The main disadvantage in using parametrics for this type of exercise is the loss of visibility into details and specifics of the estimate. It is no longer possible to identify the cost estimated for a certain task and this is a risk factor for the performing organization. A benefit of this methodology however, is to relieve the engineering specialists of some non-technical tasks and return those tasks to the estimating specialists. Still, the personnel who will perform on the contract must maintain ownership of the estimate because they are responsible for completion within schedule and cost.

Parametric estimating techniques are often used in conjunction with other estimating methods. According to DCAA, "the use of parametric estimating as the only method is considered most appropriate when the program is at the engineering concept state, or when no bill of materials exists and the program definition is unclear."[16] When it is determined that parametric estimating does not meet the criteria for a particular proposal, this method can be used to justify the cost derived using *bottom-up* techniques. This can help win high cost credibility ratings from the evaluator. Parametric models are often used by evaluators for this same purpose in their internal costing.

Just as a government auditor can use parametric models to evaluate the costs of competitive proposals, the contractor can use them to aid in source selection for subcontract work. Using relatively little input from the vendor, the models can be used to establish a *should-cost*. This can be used as a negotiation position to drive down an exaggerated cost or identify a vendor whose cost estimate is too low and will probably result in overruns.

Use of Parametric Models in Control

The quality of a company's product and service is an important consideration in awarding a contract. Each contractor needs a method of measuring themselves, both against their past performance and their competitors. The term MDQ is IBM's strategy for total quality management that focuses on developing a quality system.

A set of criteria has been developed to aid individual organizations within IBM in assessing their quality level. The assessment embodies a set of core values and concepts. Among these are the MDQ principles, continuous improvement and innovation, design quality and problem prevention, and management by fact.

MDQ is a strategic concept directed toward gaining market share, profitability, and retaining satisfied customers. It demands sensitivity to customer and market requirements and measurement of the factors that drive customer satisfaction. This includes understanding basic requirements for products and services and going beyond them to delight customers.

Continuous improvement and innovation requires a quantitative basis for assessing progress and for deriving information for future cycles of improvement. Through this, improvements in the form of providing new and improved products and services to customers, reducing errors, defects, and waste, and increasing responsiveness, productivity and effectiveness can be achieved.

Design quality and problem prevention means that strong emphasis is placed on building quality into products and services by focusing on the processes through which they are produced. MDQ focuses on process defects instead of product defects in order to intervene at the earliest possible stage.

Management by fact requires that reliable data be available for process management. Analysis is performed to extract information from this data to support decision making regarding quality assessment and quality improvement. Performance indicators are measurable characteristics of products, services, processes, and operations used to evaluate performance and to track progress. These indicators are selected to represent the factors that determine customer satisfaction and operational performance.[17]

In order to pursue MDQ as described by this assessment criteria, the concept of metrics has come into play. A metrics program is a means to collect relevant data on cost and non-cost aspects of a program. This data can then be used to evaluate a program at different stages of its life as well as compare the achievements of individual programs. The concept of metrics is integral to maintaining the integrity of the cost estimating system when parametric modeling is used. As was discussed previously, a contractors cost estimating system must be based on historical data and logical relationships.

The following is an excerpt from the keynote presentation by DCAA representative, Lawrence Uhlfelder, at the 1991 International Society of Parametric Analysts (ISPA) conference: "Cost Accounting Standards (CAS) provides guidance in accounting for contract costs at larger contractors. CAS 401 requires that a contractor's estimating practices be consistent with those governing the accumulation and reporting of costs during contract performance. Some see parametrics as being inconsistent with CAS 401 -- we do not subscribe to this. However, care must be taken to ensure both costs and noncost information used in estimating is accumulated and reported."[18]

This means that contractors may have to modify existing information systems or develop new ones in order to monitor and document non-cost variables. A metrics system can aid the contractor in meeting requirements for its cost estimating system as well as accomplish the goals of MDQ. The input data used in the parametric model, the estimate generated by the model, and the actual costs incurred can be collected and maintained. This database can then be used to draw upon for future proposal efforts and improve quality of current programs.

The series of reviews and audits FSC has in place are part of the method of achieving and maintaining customer satisfaction. Through customer meetings, the engineers and program managers can ascertain whether or not customer requirements are being met. As changes to the system architecture are made, the customer is made aware of impacts to design, schedule, cost, and other risk areas. Parametric models can be used to assess these impacts. Once an estimate has been performed, updating and revising the estimate for changes requires minimal time and effort, but can provide important insight.

The idea of DTC, or designing a system to meet customer funding requirements, has recently emerged as a valuable technique. The cost output from the PRICE model can be considered a DTC goal. As stated previously, systems are rarely specified accurately prior to contract performance. Parametric models can be used to estimate the impact of design changes on cost.

Returning to the example of the aircraft clock, consider a new customer requirement that calls for radiation hardened technology. The estimator can make a change to the platform value in the PRICE model to reflect this new specification and determine its impact. The cost ramifications may be great enough to dissuade the customer from altering the design, and this can be determined before the change is implemented.

VI. Validation and Justification

According to comments (credited to Barbara Kitchenham) made to the Constructive Cost Model (COCOMO) Users' Group, many criteria should be considered when validating a cost estimation model.[19] Some of them are discussed here. First, the input parameters should be measurable and objective. Where inputs are subjective, they will vary substantially from individual to individual. For use of the PRICE model, subjective parameters such as design experience should be clearly defined in terms of past experience to eliminate problems in assessing their values.

Second, the model should be easy to use. The difficulty and cost of obtaining input data should not inhibit the use of the tool. As was shown by the detailed example in section V, the input parameters required for a PRICE H estimate are available and easily accessible during conceptual design.

Third, the model should be general enough to be used in the different environments for which the estimator is responsible. The military environment, for which PRICE H was designed, is the main focus of the FSC cost engineering group. The model has also been useful in other areas such as commercial applications and integration efforts.

Fourth, the model should be comprehensive and include the majority of project-related activities in the estimate. The purpose of the PRICE H model is estimation of hardware and its development and production. The major activities related to FSC hardware programs are included in the activities defined in each PRICE cost category found in Appendix A. Program elements such as software and installation that have been found to be outside the scope of PRICE H can be estimated using other parametric cost models.

To justify the change to a new methodology, internal and external customers must be convinced of the superiority of the new method over the old method. It has been theorized that the parametric approach offers an improvement in terms of time, number of people involved, and cost over the traditional *bottom-up* approach. This can be proven by conducting one or more pilot programs. During a trial period, both methods should be used in parallel and a comparison of these factors made at the completion of the estimate. The resulting estimate from the parametric cost model should be compared to the estimate made from a *bottom-up* or other approach. The parametric method should be as effective as other methods at accurately predicting costs in order to justify its use.

Program Beta (the name has been changed) was used by cost engineering as a test case to demonstrate parametric estimating capabilities. The strategy that had previously been chosen for this relatively small, competitive proposal was a team approach to estimate the costs of subsystem hardware, software, and integration of these elements, and integration and installation of the subsystem into the customer's existing system. At this point, the architecture had been chosen, thereby precluding the use of parametric cost models for the planning stage of the program. Despite the program manager's initial skepticism, he allowed cost engineering to conduct a parametric analysis in parallel with the team effort during proposal preparation.

Hardware and subsystem integration costs were estimated using PRICE H. The PRICE H model is currently being incorporated into cost engineering's estimating methodology where applicable. The software parametric estimate was completed using Costar COCOMO. System integration and installation costs are beyond the scope of either model and were excluded from the study. To maintain the validity of the test, access to the *bottom-up* estimate was denied to the parametric estimators during the exercise. The results are illustrated in Table 4.

Table 4 - Project Beta Results

	ESTIMATE	
	Proposal	Parametric
NUMBER OF ESTIMATORS	6	2
TIME CHARGED	456	58
COST ESTIMATE ($K)		
Recurring Hardware	1,959	1,933
Software Development	1,529	1,767
Non-Recurring Engineering (NRE)	3,288	2,377
Support/Program Management	748	678
Integration and Test	1,261	1,103
Total Project Cost	8,785	7,858

NOTE: The program name and cost values have been altered to protect proprietary data. The relationship between the estimates is unchanged.

The parametric approach required the efforts of two estimators (one to estimate hardware, and one to estimate software) in addition to two lead engineers who provided the necessary technical inputs. The *bottom-up* approach involved six estimators (one cost engineer to estimate recurring hardware, and five others to submit budget sheets). The time required, measured in labor hours charged to B&P expense, was significantly less for the parametric approach.

The cost estimate was divided into five major categories: recurring hardware, software development, NRE, support and program management, and integration and test. The hardware estimate for both approaches included material and hands-on labor. The software estimate included development effort based on an estimate of SLOC to be delivered. For the parametric approach, NRE combined systems engineering effort from the COCOMO model with drafting, design, and systems engineering from the PRICE H model. The proposal method involved budget sheets from the systems engineering and hardware development organizations to arrive at an estimate of NRE. Parametric estimates for support, program management, and integration and test included hardware and software contributions for both approaches. Each PRICE H cost category is defined in Appendix A, taken from the *PRICE H User's Reference Manual*.[20]

The estimates at the bottom-line differed by approximately 11%. Although this result is fairly impressive, it does not provide conclusive evidence. Looking closely at the component breakout of costs reveals significant differences. Calibration of the models is required to bridge the difference between organizational assumptions found in the models and the true organization structure of the user's company. Using the model over a period of time to estimate other pilot programs and calibrating the model output to reflect past performance should provide additional substantiation for its use, and the accuracy of the estimates should improve.

Submission of the parametric cost estimate can serve as cost justification for the *bottom-up* approach if it is within a certain range of acceptability. In this case, the Beta program manager was impressed with the results of the parametric estimate and included it in the winning proposal submission to prove cost realism. In addition, this served as an opportunity to gain insight into unanticipated concerns and problems from the customer during the review process. If the new approach will not be ultimately accepted by the customer, there is no value in implementing it as a proposal preparation tool.

The assumption of this test of validity is that the accuracy of the *bottom-up* method has been verified through historical use. A good *bottom-up* cost estimate is generally expected to be within 5-10% of the actual cost. Therefore, the Beta test case would be considered successful, knowing that the model had not yet been calibrated for this estimate. The comparison of estimated costs before performance on the contract is completed is not sufficient. In addition, the pilot program should be extended through performance to determine the adequacy of the parametric method as a predictor of cost. At this point, performance on Beta has not been completed, therefore no conclusions can be drawn as to the accuracy of the estimates.

Another possible test would be to perform parametric cost estimates for programs that have been completed and for which actual costs are known. In this way, the accuracy of the estimates can be evaluated in the present, without having to wait until the program is completed. The availability of necessary parametric inputs for a historical program can be a limiting factor. In addition, during the life of a program, many engineering change proposals (ECPs) may have been incorporated that will affect the inputs and the actual costs incurred. Even if the input data can be obtained, this has only limited usefulness because skeptics will believe the estimator *backed into* the actual costs.

As mentioned previously, the specification of variables and estimating procedures and assumptions inherent in the model are unknown. Therefore, analysis of the model's sensitivity to changes in attribute values will lead to an understanding of the cost estimating relationships. Sensitivity analysis of physical, personnel, and project attributes in the PRICE model was performed using Program Beta.

The system in the Beta proposal included five individual hardware units. Physical attributes such as quantity and weight were varied to show the impact on cost outputs. The weight of each unit (represented on the graph as x) was varied to 50%, 75%, 150%, and 200% of x to test the sensitivity to errors in weight estimates. The output, as illustrated in Figure 13A, shows a significant impact to all cost categories. The cost output is defined in terms of NRE (including drafting, design, systems engineering, project management, and data) and hardware (including manufacturing and tooling and test equipment) costs.

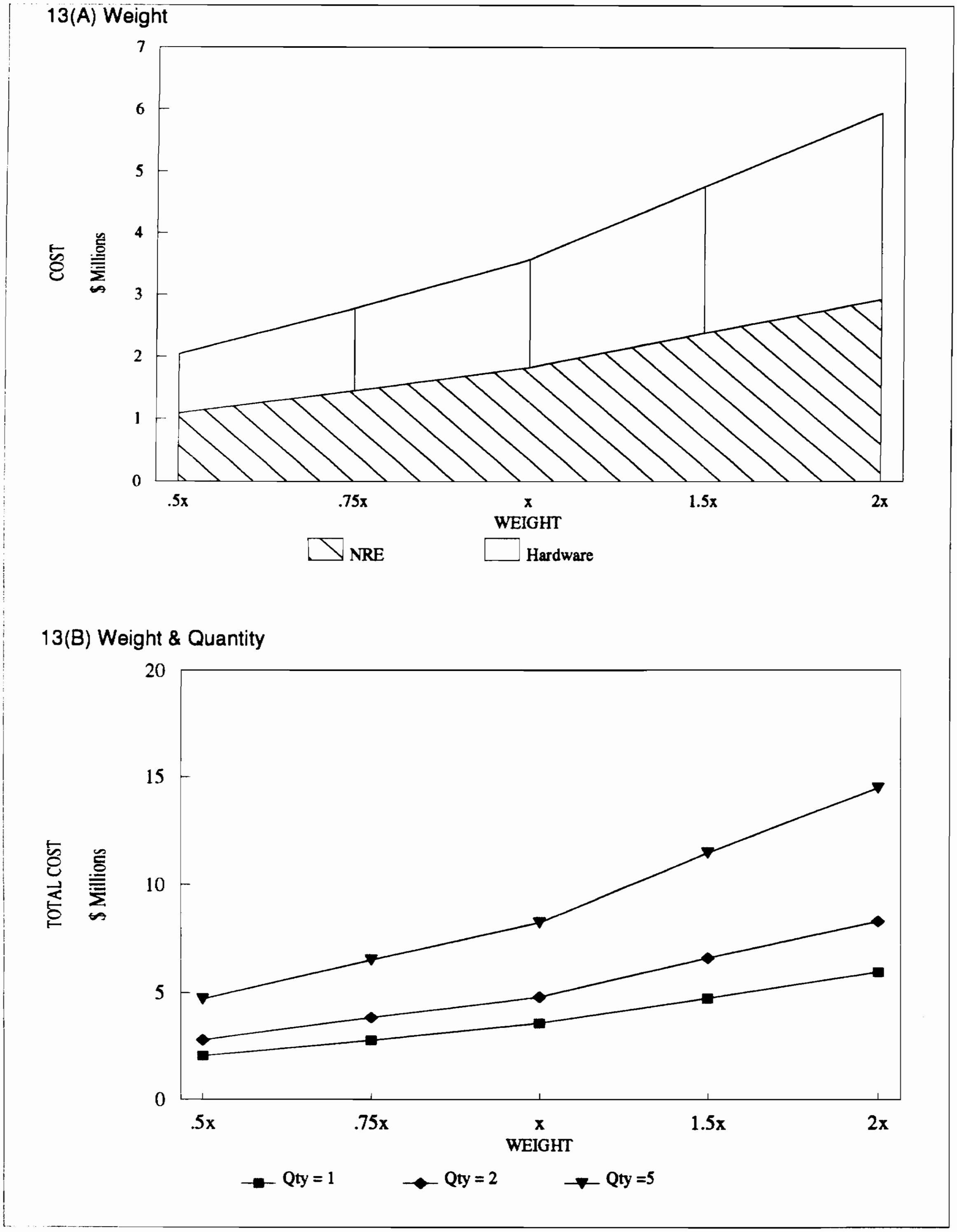
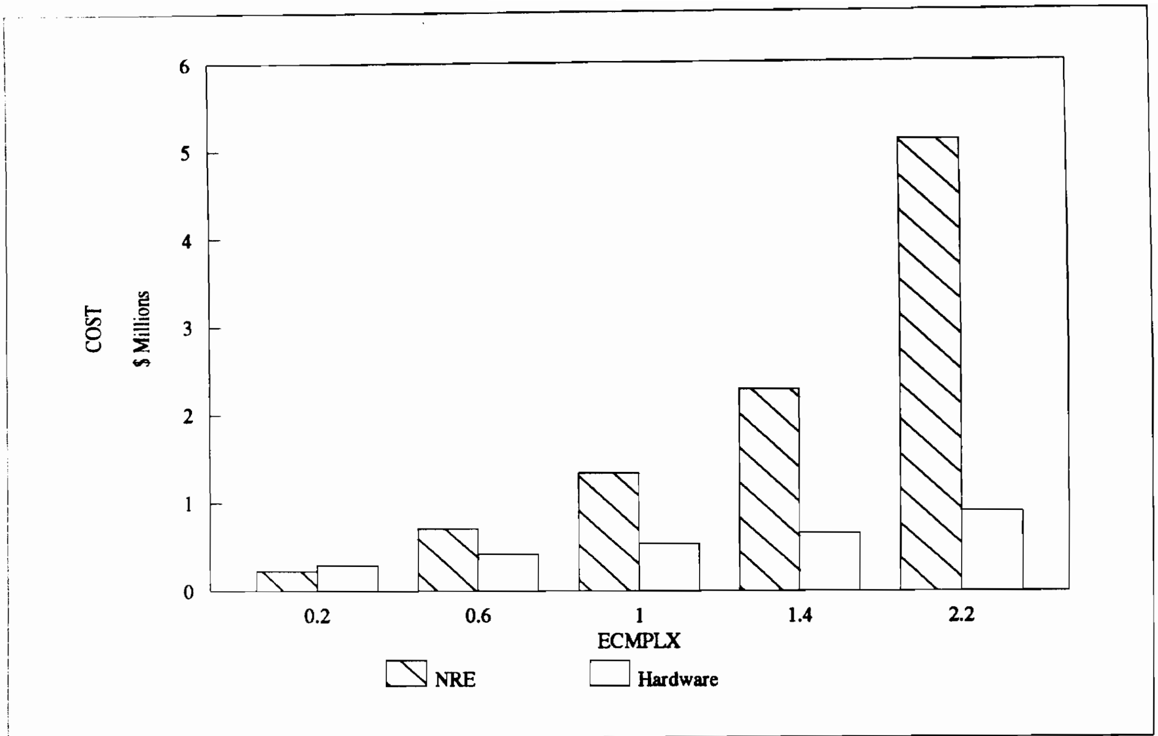


Figure 13 - Sensitivity Analysis of Physical Attributes (Quantity & Weight)

The total costs are shown in Figure 13B for different quantities of prototype systems (one, two, and five). While the increase in total cost is obvious, an expected decrease in unit cost due to quantity production is also apparent. This is due to the fact that the model assumes that each prototype is not being designed and manufactured independently.

Personnel attributes were examined using the engineering complexity (ECMPLX) variable. ECMPLX is a combination of design team experience and design scope. A qualitative description of each value is provided in Figure 14. In addition, Figure 14 illustrates the impact of changes to ECMPLX values on cost outputs for one unit of the Beta system.

The analysis confirms what one would intuitively expect, that is, a dramatic impact to the engineering costs and a much lesser impact to the manufacturing costs. The model accounts for engineers' experience in completing similar types of design. It also indicates a slight difference in manufacturing costs due to the experience across previous projects in completing hands-on labor.



Key to Engineering Complexity (ECMPLX) values:

0.2 Extensive experience,
Simple Modification to an existing design

0.6 Extensive experience,
Extensive Modifications to existing design

1.0 Normal experience
New design within established product line

1.4 Mixed experience
New design different from established product line;
utilizing existing materials and electronic components

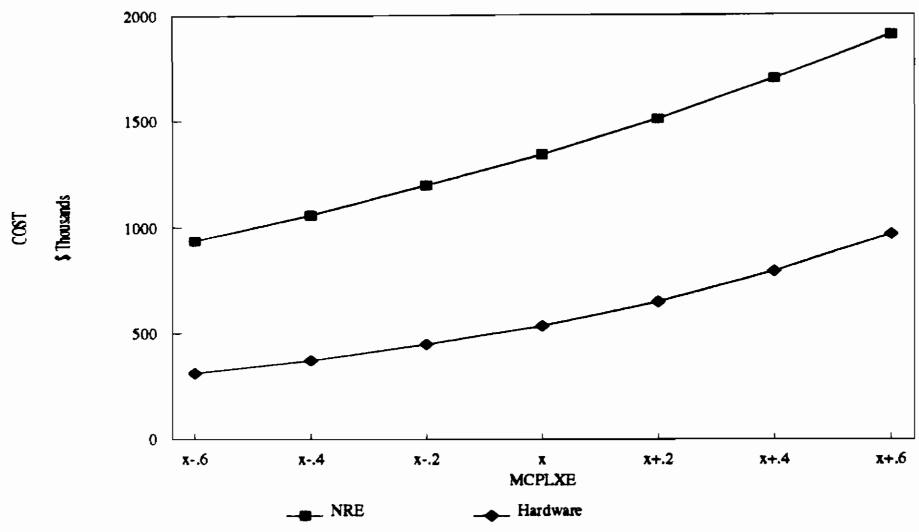
2.2 Unfamiliar with design, many new to job,
New design different from established product line;
requires in-house development of materials or components

Figure 14 - Sensitivity Analysis of Personnel Attributes (ECMPLX)

Various project attributes were analyzed, including: manufacturing complexities (MCPLXE and MCPLXS), platform (operating environment), and percent of new electronics design. The sensitivity of manufacturing complexity values is illustrated in Figure 15. In Figure 15A, MCPLXE values are iterated by .2 from the original value, x. This shows a clear relationship between cost and the difficulty of designing and manufacturing the hardware.

Similar changes to MCPLXS values, shown in Figure 15B, do not appreciably impact either cost category. This shows the model is not significantly sensitive at the low end of MCPLXS values (where Beta exists). The NRE costs can be impacted by increasing the new structure design (NEWST) from 0% to 50%. This does not change the manufacturing (hardware) cost (the two lines are superimposed).

15(A) - Manufacturing Complexity of Electronics (MCPLXE)



15(B) - Manufacturing Complexity of Structure (MCPLXS)

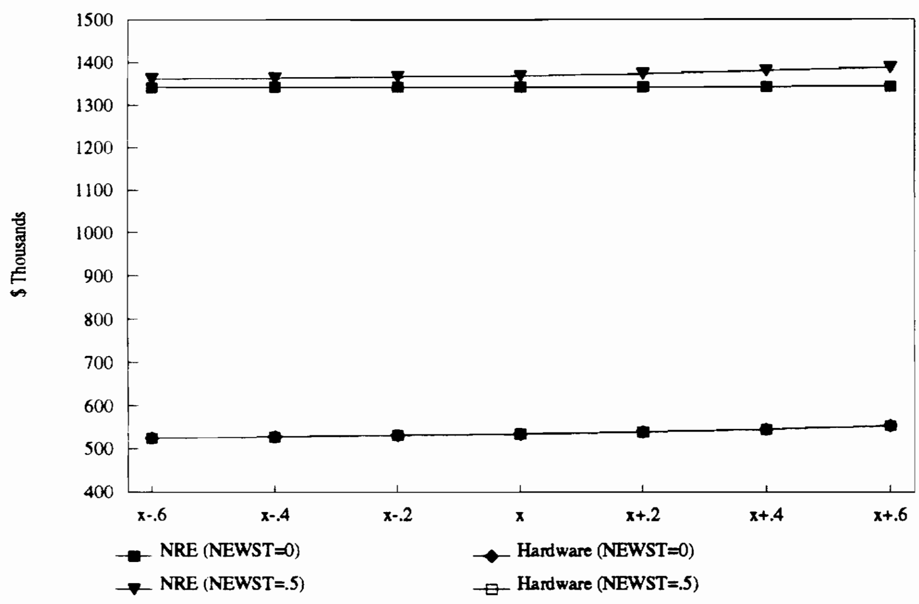


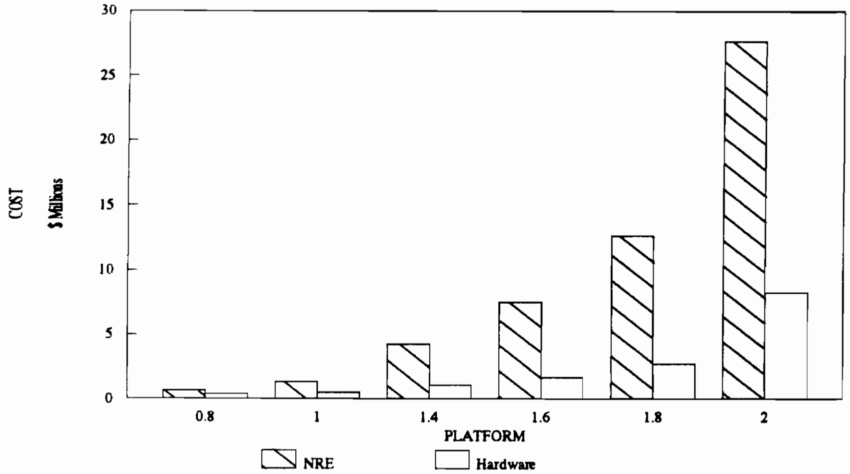
Figure 15 - Sensitivity Analysis of Project Attributes (MCPLXE & MCPLXS)

The results of tests on platform values are illustrated in Figure 16A. Variations in platform values, from a commercial environment on one end to unmanned space on the other, show a significant relationship to NRE costs. This indicates that, as operating and reliability constraints increase, the cost to design hardware for those environments also increases. The platform value does not directly impact hardware cost. However, the manufacturing complexity values are also a function of platform and it is this change that causes an increase in the cost to manufacture the hardware. Based on this, the analysis indicates a significant change to hardware costs across operating environments.

Changes to percent of new electronics design, illustrated in Figure 16B, also indicate a direct relationship to cost for NRE, but has no impact on manufacturing hardware costs.

In addition to considering the time and cost savings and accuracy obtained through the use of parametric cost models, implementation costs should also be considered. Costs involved with equipment, licenses, operation, and training for a commercial model must be weighed against proposed benefits.

16(A) - Platform



Key to PLATFORM values:

0.8-Commercial
1.0 - Ground
1.4 - Mobile
1.6 - Submarine
1.8 - Military Air
2.0 - Unmanned Space

16(B) - % New Electronics Design (NEWEL)

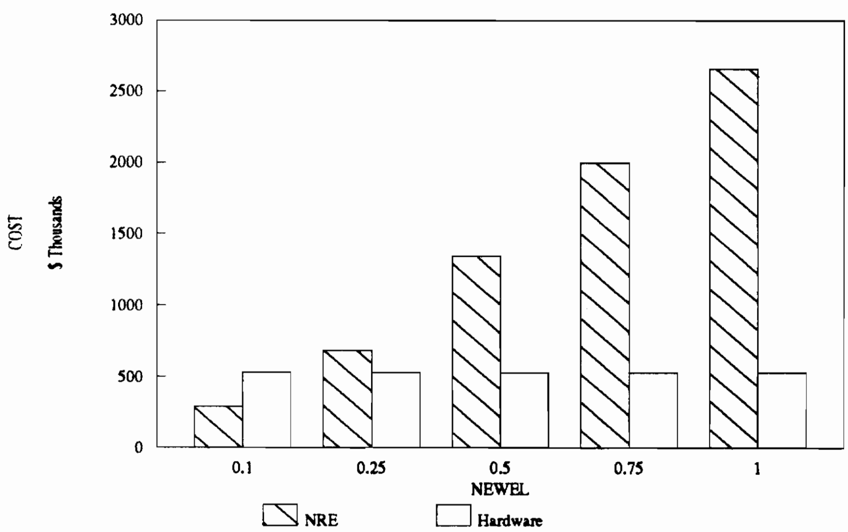


Figure 16 - Sensitivity Analysis of Project Attributes (Platform & NEWEL)

Aside from these cost considerations, there are several non-cost issues that should also be addressed. The introduction of a parametric estimating process will impact the jobs in various areas across the site. The most obvious change will be to the cost engineering organization. Estimators who are accustomed to using a *bottom-up* approach will be given the opportunity to broaden their scope of estimating responsibility. Where they were previously responsible for estimating only recurring hardware costs, they will now have the ability to estimate costs for entire programs. In order to do this, training will be required to learn this new skill. The perceived benefits of these changes will be based on individual preferences and circumstances.

The performing organizations who traditionally estimate their own effort and submit budget sheets will be relieved of the estimating responsibility. The perceived benefit of this will again vary from individual to individual. The process by which these organizations maintain visibility and control of the estimates will help determine the acceptance of the change.

There is also an impact to other outside organizations aside from estimating responsibilities. As was discussed in Section V, the cost collection methods must be consistent with estimating methods. Therefore, the financial systems may need to be revised to collect cost data in the appropriate format as well as to collect non-cost data. This will represent a significant change to the jobs performed by the financial program control organizations. The pricing group is responsible for taking the proposal cost inputs and incorporating adders such as cost of money and fee for final submission to the customer. The format of the inputs will be altered, which will affect their systems and processes.

Another non-cost factor that should be considered is the time and effort required to implement the new process. Cost engineering in Manassas has been working for the past 18 months to introduce the idea of parametric cost estimating and gain acceptance of its use. Focus for the following months will be placed on calibrating the model and setting up a metrics program to collect cost and non-cost data. Estimators and management must be dedicated to using the new approach to warrant the substantial effort that is required.

VII. Conclusions and Recommendations

Use of parametric cost models in the conceptual and planning stages of a contract is valid today, without calibration, as shown by the example in section V. In order to make comparisons between solution alternatives, a common *basis of estimate* is needed and parametric models can provide this. Models are also an effective cross-check of analyses of customer funding and potential competitors made through other sources. A commercial model provides the cost of a theoretical norm in industry, which is useful as a benchmark. Changes made during the conceptual design stage can be modeled fairly quickly to estimate impacts to cost and schedule and the model output can be used to influence decisions.

Care should be taken when using parametric cost models in the estimating phase for proposal preparation. Calibration of the model is necessary to meet the requirements of FAR as discussed in Section V. Enough historical data must be collected in the required format to show a logical relationship between cost and non-cost variables. This can not be completed adequately at the time of proposal preparation. A metrics program geared toward the model inputs must be in place to collect the data on similar programs. "Similar" is the key here; a program chosen for comparison must be proven to be similar. A great deal of time for planning and implementation of a metrics program is necessary to make this effort productive. Once a model has been calibrated, it can be used to prepare an estimate for proposal, or verify and validate an estimate performed by a *bottom-up* method.

Parametric cost models can also be used in the control phase of a program to influence and aid quality and risk assessment. With this method, dissimilar programs can be compared and progress can be monitored throughout a program. The impact of design changes on cost and schedule can also be assessed efficiently. However, parametric cost models are not cost-collection tools. The data maintained through a metrics program is a necessary element of incorporating the use of these models.

Another alternative to the problems presented here would be to create a *home-grown* model for the specific company, organization, or task for which an estimate is required. A *home-grown* model would reflect IBM business practices because it would be based on IBM historical data. In this way, the need for calibration of the model would not be required. It also guarantees a relationship between the variables that is supported by fact.

Developing a *home-grown* model would provide an opportunity to study and understand the relationships unique to an organization. This is an important benefit which may outweigh the time and cost necessary for implementation. This level of understanding can not be obtained for a commercial model unless the estimating relationships are made available to the users.

However, there are several problems associated with this approach. The most significant of these is the need for reliable historical cost data. Statistical analysis is required to ensure that relationships are logical and not based on chance occurrence. Historical data that is readily available is not generally in the format necessary to perform this analysis. This is due to the varied formats of WBSs and reporting requirements of the different contracts.

The time and effort required to establish non-parametric relationships may be prohibitive. In addition, it may be difficult if not impossible to get agreement on the important cost drivers without including every variable that influences cost. Potential benefits of creating an IBM model could be negated by high development costs.

Using an existing, commercial parametric cost model offers some advantages over this option. The cost relationships embedded in a commercial model are established and tested by industry. Certain models, such as PRICE, are recognized and accepted by government auditors, and in many cases, they are being used to do proposal evaluations. A parametric model uses a minimum of non-cost variable inputs so that the time required to perform a program estimate is much less than that required for a *bottom-up* estimate. All costs, including both recurring and non-recurring costs can be estimated with one set of inputs, whereas numerous models or estimators would be needed for other approaches.

A basic premise is that the CERs in a commercial model are correct. There is no way for an estimator to verify this. In addition, testing is required to prove the statistical significance of the variables chosen to predict cost. For instance, the significance of weight as a variable affecting cost has already come into question.

PRICE H may or may not be the appropriate tool for parametric cost estimating in another organization. Its usefulness must be determined on an individual basis according to the criteria previously discussed. In addition, the advantages and disadvantages of implementing a new process must be considered based on applicability of parametric cost estimating to other situations.

VIII. Notes

1 *Systems Engineering Principles and Practices* (*SEPP*), IBM, Federal Sector Division Training Course, December 1991, p.SE-2.

2 *Systems Engineering Management Guide* (*SEMG*), Defense Systems Management College, December 1990, p.1-2.

3 *SEPP*, p.DC-3.

4 *SEPP*, p.SE-6.

5 *SEPP*, p.SE-2.

6 *SEPP*, p.AD-3.

7 Blanchard, B.S., and W.J. Fabrycky, *Systems Engineering and Analysis*, Prentice Hall, 1990, p. 551.

8 *IBM Federal Systems, Technical Management Plan, Development Guidance*, 1987.

9 *SEMG* p.15-1,2.

10 *Technical Risk Assessment*, Department of the Navy, March 1986, p.1.

11 *SEMG*, p.17-1.

12 *Federal Acquisition Regulation* (*FAR*), Section 215:811-70 Definitions.

13 *FAR*, Section 215:811-76 Characteristics of an Adequate Estimating System.

14 *Journal of Parametrics*, International Society of Parametric Analysts (ISPA), Volume XI, Number 1, August 1991, pp.5-7.

15 *Journal of Parametrics*, August 1991, p.8.

16 *Journal of Parametrics*, August 1991, pp.11-12.

17 *IBM Market Driven Quality Assessment and Recognition 1992 Guidelines*, February, 1992, p.6.

18 *Journal of Parametrics*, August 1991, p.10.

19 *The Sixth International COCOMO Users' Group Meeting*, Software Engineering Institute, October 23, 1990.

20 *PRICE H User's Reference Manual*, General Electric Company, 1990, p.5-4.

IX. Glossary of Terms

ACWP - Actual Cost of Work Performed

BAFO - Best and Final Offer
B&P - Bid and Proposal
BCWP - Budgeted Cost of Work Performed
BOM - Bill of Materials

CAS - Cost Accounting Standards
CDRL - Contract Data Requirements List
CER - Cost Estimating Relationship
CLIN - Contract Line Item
COCOMO- Constructive Cost Model
CPI - Cost Performance Index

DCA - Document Coordination and Authorization
DCAA - Defense Contract Auditing Agency
DoD - Department of Defense
DTC - Design to Cost

EAC - Estimate at Completion
ECP - Engineering Change Proposal

FAR - Federal Acquisition Regulation
FSC - Federal Systems Company

IBM - International Business Machines Corporation
ILS - Integrated Logistics Support
IR&D - Independent Research and Development
ISPA - International Society of Parametric Analysts

LCC - Life cycle cost
LTD - Live Test Demonstration

MCPLXE- Manufacturing Complexity of Electronics (PRICE)
MCPLXS- Manufacturing Complexity of Structure (PRICE)
MDQ - Market Driven Quality
MTM - Motion Time Measurement

NRE - Non-Recurring Effort

PRICE - Parametric Review of Information for Costing and Evaluation

RFP - Request for Proposal

SEMP - Systems Engineering Management Plan
SLOC - Source Lines of Code
SOW - Statement of Work

TMP - Technical Management Plan

WBS - Work Breakdown Structure
WBSID - Work Breakdown Structure Identification

Appendix A PRICE Cost Elements

Cost Category	PRICE Includes:
Drafting	Manufacturing drawings Data lists Specifications documentation Incorporation of engineering changes
Design	Design engineering Laboratory experimental work Breadboarding and testing Specifications design
Systems Engineering	Effort to convert performance requirements into design specifications
Project Management	Program management and control Travel and living expenses Reliability, maintainability, quality assurance Computer operation costs Preparation of in-house reports
Data	Operations and maintenance manuals Spares lists Deliverable drawings Status reports Contract data requirements list (CDRLs)
Recurring Hardware	Material Assembly and test labor costs Qualification test costs Quality control and line inspection Set-up costs
Tooling and Test Equipment	Special tools Special test equipment