

2011

**University of North Carolina Wilmington
Master of Science in
Computer Science and Information Systems
Proceedings**

<https://csbapp.uncw.edu/mscsis>

Comparative Assessment of Software Development Project Effort Estimation Tools

Lorie McClain

A Capstone Project to be Submitted to the
University of North Carolina at Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Masters of Science

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2011

Approved by

Advisory Committee

Dr. Drew Rosen

Dr. Devon Simmonds

Dr. Thomas Janicki

Abstract

Despite the fact there has been ongoing research on software development project effort estimation techniques for over forty years, studies have shown that, on average, projects consistently overrun the original effort estimates in time and dollars. These costly overruns have left many organizations in a situation where the only alternatives are to abandon a project or to start over with a new project which leaves the project objectives unsupported for much longer than intended. Software development projects have become increasingly maligned and the delivery methods are under heavy suspicion by those who provide the funding. The traditional estimation methods which are based on either expert judgment or formal mathematical models are not proving to be as reliable as needed and there is debate as to the best method to use to achieve maximum accuracy. This paper provides an assessment of actual project data against both expert estimations and formal models via use of an automated tool. Results indicate that the expert estimation method is more accurate than the formal methods, however, there are noted shortcomings with the research so guidance is provided to increase the validity of any future investigation in this area.

List of Tables & Figures

Table 1 - A comparison of the processes for expert-based and model-based estimation 9

Table 2 - List of Toolsets with Preliminary Feature Review14

Table 3 - Additional Toolset Preliminary Feature Review19

Table 4 - Brief Description of Sample Projects24

Table 5 - Project Attributes24

Table 6 - Summary of Usability Ratings26

Table 7 - Comments on Usability Ratings27

Table 8 - Summary of Additional Usability Ratings28

Table 9 - Summary of Effort Level for Inputs29

Table 10 - Project 1 Effort Estimates versus Actuals; no adjustments31

Table 11 - Project 1 Effort Estimates versus Actuals; adjusted for training and SME hours32

Table 12 - Project 2 Effort Estimates versus Actuals; no adjustments32

Table 13 - Project 2 Effort Estimates versus Actuals; adjusted for training and SME hours33

Table 14 - Project 3 Effort Estimates versus Actuals; no adjustments34

Table 15 - Project 3 Effort Estimates versus Actuals; adjusted for training and SME hours34

Table 16 - Project 4 Effort Estimates versus Actuals; no adjustments35

Table 17 - Project 4 Effort Estimates versus Actuals; adjusted for training and SME hours35

Table 18 - Project 5 Effort Estimates versus Actuals; no adjustments35

Table 19 - Project 5 Effort Estimates versus Actuals; adjusted for training and SME hours36

Table 20 - Project Totals in Hours; with and without adjustments for training and SME hours37

Table 21 - Design and Development Totals in Hours38

Table 22 - Project Management Totals in Hours38

Table 23 - Requirements Totals in Hours38

Table 24 - Deployment Totals in Hours39

Table 25 - Testing Totals in Hours39

Table 26 -Totals in Hours by Project Type39

Table 27 -Totals in Hours by Project Length in Months40

Table 28 –Totals in Hours by Project designed for Internal or External Use.....	40
Table 29 –Totals in Hours by Development Methodology	40
Table 30 –Totals in Hours by Technical Classification	41
Figure 1 – Planned versus Actual Capstone Project Timeline	47

Contents

1. Introduction	6
1.1. Problem Statement.....	6
1.2. Proposed Solution.....	6
2. Brief Overview of Software Development Project Effort Estimation Techniques	8
3. Potential Advantages and Pitfalls of Use of Formal Estimation Tools	12
3.1. Potential Advantages	12
3.2. Potential Pitfalls	13
4. List of Toolsets Reviewed	14
5. Research Process	20
5.1. Exploration of inputs and outputs for each toolset	20
5.2. Historical project data for research use	23
5.3. Project Manager interviews.....	24
5.4. Generate Algorithmic Project Estimates.....	25
6. Analysis	26
6.1. Subjective analysis.....	26
6.2. Objective analysis	30
7. Summary of Conclusions and Recommendations	42
7.1. Summary of Conclusions	42
7.2. Recommendations for Future Research	42
7.3. Recommendations for an Improved Project Effort Estimation Tool	43
8. Works Cited.....	45
Appendix A – Estimation Models Referenced in PRODUCT H Documentation.....	46
Appendix B – Planned Capstone Project Timeline versus Actual Capstone Project Timeline	47
Appendices C, D, and E removed to protect proprietary information	48

1. Introduction

1.1. Problem Statement

Research has shown that software development projects overrun original effort estimates, in terms of hours, by 30 – 40% on average. This same research has also concluded that most organizations create their project effort estimates based on an assessment of project scope and complexity as provided by expert individuals within the organization. (Jorgensen & Molokken, 2003). Because there are a number of risks associated with use of expert estimation, it has been suggested that organizations might want to consider a combination of techniques, including both expert judgment and the use of a mathematical model (Blattberg & Hoch, 1990). Multiple estimation results could then be used to validate or challenge assumptions and, perhaps, allow the organization to hone in on a more accurate final value.

Based on discussions with project managers in an organization which solely uses expert judgment for creating software development estimates, the author believes these organizations will be reluctant to add a formal method to the estimation process unless the method is intuitive, requires input that is relatively easy to obtain, and does not require excessive set-up or maintenance. However, to gain credibility, the model must also either prove to be at least as accurate as the expert method or show value by providing an objective “jumping off point” for expert estimation.

1.2. Proposed Solution

Formal estimation methods have been developed, compared and researched for over forty years. A number of these methods have been used to create commercially available toolsets. Some of these toolsets are based on models with a fully disclosed set of mathematical calculations. Others are based on proprietary algorithms developed through review of historical datasets culled from numerous years of project data.

In an attempt to arm organizations with decision-enabling data, this research has:

1. Identified a list of potential toolsets for review

2. Performed a feature assessment targeted at identifying toolsets which appear to meet the criteria of (a) intuitive, (b) requiring easily obtainable input, and (c) not requiring excessive administrative maintenance. Select a subset of these tools for further investigation.
3. Evaluated the usability of the selected tools using a sample of project data available from actual completed projects.
4. Evaluated the results of the selected tools by comparing their outputs against each other and against the historical results from the sample of projects.
5. Provided recommendations and guidelines for IT Project Managers when selecting an estimation tool.

2. Brief Overview of Software Development Project Effort Estimation Techniques

Software development project effort estimation is based on a prediction of effort by resource required to design, develop, test and deploy software. Once created, the project effort estimate can then be used as a baseline for a project plan. In addition, when effort by resource is multiplied by an average or actual cost rate, it can be used for organizational budgeting, investment analysis, or proposal bidding purposes. The general steps required for project estimation can be summarized as follows (Jorgensen, Boehm, & Rifkin, 2009):

1. Define the estimation problem
2. Agree on decisions and assumptions relevant for estimation
3. Collect data required for the estimation
4. Weight the importance of various types of data
5. Quantify the effort based on the data
6. Review and refine the results.

Although the steps appear straightforward, the ability to accurately estimate the effort required for a software development project is a serious weakness in the software development process. One of the issues with the process is the quantification step which is generally based on incomplete and/or uncertain input. A second issue is the choice of methodology used to quantify the effort based on the selected input. A third issue is the noted lack of algorithm “tuning” based on comparisons to actual results over time.

Since as early as the 1960s, researchers have been struggling with estimation method weaknesses and attempting to develop a repeatable, reliable process to improve software development projections (Cuadrado-Gallegoa, Rodriguez-Soriaa, & Martin-Herreraa, 2010). The research has led to a variety of estimation techniques which have been generally categorized as either expert judgment method or formal model method. The most notable difference in the models is the nature of the quantification step. In expert judgment estimation, this step is based on subjective processes and knowledge. In formal model estimation, this step is based on a prescribed process, generally derived from historical data analysis. The table below summarizes

the variations in the processes based on the nature of the chosen quantification technique (Jorgensen et al., 2009).

Table 1 - A comparison of the processes for expert-based and model-based estimation

Estimation activity	Expert judgment	Estimation models
Understand the estimation problem.	Judgment- and analysis-based processes possibly structured by templates for work breakdown.	Judgment- and analysis-based processes possibly structured by the need for model input if the model has already been chosen at this stage.
Agree on decisions and assumptions relevant for estimation.	Judgment- and analysis-based processes possibly structured by checklists or guidelines.	Judgment- and analysis-based processes possibly structured by checklists or guidelines. Several assumptions implicitly made when selecting a model.
Collect information relevant for the estimation.	Judgment- and analysis-based processes possibly structured by checklists or guidelines. The set of variables considered relevant is judgment based. The collected information includes subjective assessments.	Judgment- and analysis-based processes structured by the model's need for input data. The set of variables considered relevant is predetermined. The collected information typically includes subjective assessments.
Evaluate the importance (weighting) of different pieces of information.	Judgment-based processes.	Analysis-based processes based on statistical analysis of historical data.
Quantify the effort on the basis of the information.	Judgment-based processes.	Analysis-based processes based on a formalized, repeatable process or formula.
Review the effort estimate.	Judgment- and analysis-based processes possibly structured by checklists, guidelines, and expert judgment.	Judgment- and analysis-based processes possibly structured by checklists, guidelines, and expert judgment. Sometimes results in judgment-based updates of the effort estimate.

According to an evaluation conducted in 2004, the majority of historical academic research in software development estimation has been concentrated on attempts to construct and validate software effort estimation models. Of those models, 83% have been formal estimation models (Jorgensen & Shepperd, 2007). Formal estimation models are generally parametric models or analogy-based as described below:

1. Parametric estimation (Meli & Santillo, 1999)
 - a. A model is developed containing a set of related mathematical equations which require variable parameters (e.g. lines of codes, function points, use cases)
 - b. The model is based on historical analysis of actual projects.
 - c. A software development scenario is defined by selecting a value for each parameter.
 - d. The development effort estimate is based on the results returned from the calculations.
 - e. Varieties of parametric methods include neural networks, Bayesian statistics, and regression trees
2. Analogy-based estimation (Chemuturi, 2009)
 - a. Organization must have a reasonable number of historical projects, detailed record-keeping, detailed variance analyses, a knowledge repository, and trained estimators.
 - b. All projects must be categorized according to type (e.g. development, implementation, porting), domain, functionality complexity, and various platform parameters (e.g. tiers, language).
 - c. A historical project is selected from the project repository to act as a base estimate.
 - d. A new estimate is extrapolated from the base estimate as needed to conform to any significant parameter differences.

In contrast to the formal estimation models, expert judgment models do not rely on empirical data. Experts, who have presumably participated in similar past projects, provide subjective estimates of required effort. Academic research on expert judgment methods have been relatively minor compared to the research on formal methods (Jorgensen & Shepperd, 2007). However,

there have been some improvements in expert judgment models over past years with increasing use of the Delphi Technique and work break-down estimations as described below:

1. Delphi Technique (Boehm, Abts, & Chulani, 2000)
 - a. A group of experts is asked to provide estimates without consulting other group members.
 - b. Answers are collected, organized, and returned to each member who then makes a second assessment with the knowledge of their peers' opinions.
 - c. Additional iterations can be performed as needed. The desired effect is to guide the group consensus to a reasonable result.
 - d. A variation called the Wideband Delphi Technique which allows for group interaction between estimation rounds.
2. Work Breakdown Structure (WBS) (Boehm et al., 2000)
 - a. Project elements or tasks are organized into two hierarchies, one representing the product and one representing the activities required to build the product. The hierarchical structure components are based on expert judgment.
 - b. Checklists and guidelines are generally provided to ensure important project elements are not overlooked.
 - c. Costs and probabilities are assigned to the individual elements, again based on expert judgment.
 - d. Total project cost is the summation of costs of the individual elements.

3. Potential Advantages and Pitfalls of Use of Formal Estimation Tools

3.1. Potential Advantages

Formal estimation tools have not proven to be widely used for real-life estimation exercises (Jørgensen, 2004). Most organizations continue to use expert judgment methods in spite of the fact that neither method has proven to be particularly accurate (Jørgensen et al., 2009). However, formal estimation tools have advantages in very specific circumstances that should not be overlooked. Those circumstances can include:

1. In organizations where the model is based on the organization's specific purpose or on data from a similar industry and purpose,
2. In organizations where there is a noted lack of expertise in general or for the specific software type or user domain,
3. In organizations where the experts' estimates may be challenged and negotiated for reasons other than estimate accuracy (e.g. proposal pricing),
4. In organizations where the experts may provide overly optimistic or pessimistic estimates based on "selfish" reasons such as ability to meet performance objectives or to win at office politics,
5. In organizations where the environmental change is rapid and expertise becomes dated very quickly.

In situations such as the ones above, the ability to base estimates on formal models provides the organization with a repeatable, clearly defined and well understood level of effort that can be explained and defended for both internal and external purposes as needed. And, over time, historical data can be captured from the model to allow for refinement of both the mathematical calculations and of the modeling process. Quite the contrary, expert judgment tends to provide data that is based on experience mixed with intuition which is non-repeatable, inappropriate as a basis for training for less experienced estimators, and highly susceptible to human error.

3.2. Potential Pitfalls

Clearly, if organizations are not flocking to use formal estimation tools, there must be a downside to these methods. Some noted reasons for organizational resistance to formal estimation methods have been stated as follows:

1. Input values into the models have been perceived as subject to opinion
2. Input values have been deemed to be difficult and/or time-consuming to determine
3. Individuals may be reluctant to define measurable values for fear those values may become productivity measures which affect personal performance assessments.

Some organizations may not be reluctant to embrace a model but may still have valid reasons to prefer an expert judgment model. Formal estimation models may just not meet organizational needs in situations where (Jorgensen, 2007):

1. Formal estimation models are not tailored to the specific industry, domain, or software context
2. Input relationships have proven to be unstable
3. Information deemed of high importance is not included in the model

4. List of Toolsets Reviewed

Seven toolsets were reviewed as possible candidates for this project. The preliminary list of tools and promised associated features are outlined below in Table 2.

Table 2 - List of Toolsets with Preliminary Feature Review

Product	Features
PRODUCT A	Model: <ul style="list-style-type: none"> • Proprietary, sector-specific parametric models derived from project histories, metrics, and behavioral models • Generates a rough estimate in minutes which can then be refined to meet staffing and scheduling constraints • Generates costs based on project-specific rates provided by user
	User Experience: Pre-defined project template or guided wizard to create a customized project
	Integrations: APIs are provided for custom integrations
PRODUCT B	Model: <ul style="list-style-type: none"> • Model is calibrated to represent the current environment by interpreting a historical database of past projects. • An information model is created by collecting software characteristics, personal attributes, and computer attributes etc. • Level of effort is calculated using an algorithm based on lines of code
	User Experience: <ul style="list-style-type: none"> • Requires fewer parameters than some other formal models • Provides pre-defined templates • Utilizes an industry knowledge base • Eliminates updating and double entry by providing smart components

Product	Features
	<p>Integrations:</p> <ul style="list-style-type: none"> • MS Office • web applications
<p>PRODUCT C</p>	<p>Model:</p> <ul style="list-style-type: none"> • Utilizes COCOMO II model delivered as an Excel add-in • Allows for multiple estimates in one Excel workbook • Accommodates multiple subsystems per estimate and multiple modules per subsystem • Supports Pert probability cost distribution: using minimum, most likely, maximum size/cost.
	<p>User Experience:</p> <ul style="list-style-type: none"> • Can perform what-if scenarios using various sizing methods • Excel user interface provides flexibility for graphing, reporting, distribution, etc.
	<p>Integrations:</p> <p>None beyond native Excel capabilities</p>
<p>PRODUCT D</p>	<p>Model:</p> <ul style="list-style-type: none"> • Proprietary model combining industry and in-house historical data • Model considers calculations of size, complexity, operating environment, • Can estimate based on any of the following: <ul style="list-style-type: none"> ○ Use Cases ○ Function points ○ Predictive object points ○ Functional size ○ Source Lines of Code (SLOC) ○ Fast function points ○ Analogies

Product	Features
	<p>User Experience:</p> <p>Provides large number of real-time metrics to allow for thorough analysis during estimation process</p> <hr/> <p>Integrations:</p> <ul style="list-style-type: none"> • MS Excel, • MS Word • MS SQL Server Reporting Services
PRODUCT E	<p>Model:</p> <ul style="list-style-type: none"> • Allows for choice of formal proprietary methods or custom estimation methods <ul style="list-style-type: none"> ○ CoCuSi proprietary model ○ Productivity rate-based model ○ Function points based model ○ COCOMO-II based model ○ Extended 3 points based model ○ Analogy model • Includes outsourcing cost model as subset of all other modules • Allows for use of one or more estimation models together • Can model and configure new estimation projects from previous projects • Can assemble estimation data from stored components • Provides Bottom-up and Top-down estimates side-by-side <hr/> <p>User Experience:</p> <ul style="list-style-type: none"> • Allows for quick estimates based on available information • Allows for generic resource Planning • Can provide MS Project Schedule generation • Utilizes MS Excel and SQL Server for ease of use

Product	Features
	<p>Integrations:</p> <p>MS Access ADP for analysis and reporting; external tools</p>
PRODUCT F	<p>Model:</p> <ul style="list-style-type: none"> • Includes COCOMO 81 and COCOMO II models • Includes a complete Jensen Model (Product A) • Includes a Norden-Putnam-Rayleigh (NPR) Model • Allows for user-extensible project categories and development profiles • Allows the user to specify uncertainty associated with the independent variables • Determines probability distributions associated with the dependent variables
	<p>User Experience:</p> <ul style="list-style-type: none"> • Full-featured drag and drop GUI • Includes freeform text notes logging for documentation of rationale • Includes interactive Ross and Gant charts • Includes staffing and metrics reports
	<p>Integrations:</p> <p>Integrations possible via XML</p>

Product	Features
PRODUCT G	<p>Model:</p> <ul style="list-style-type: none"> • Model building tool consisting of a structured format • Calculation engine including <ul style="list-style-type: none"> ○ Data normalization ○ Inflation handling ○ Time Phasing ○ Learning Curve Analysis ○ Estimate documentation ○ Risk Analysis ○ Error-trapping
	<p>User Experience:</p> <ul style="list-style-type: none"> • Multiple wizards to guide usage • Pick Lists to easily select the appropriate items • Capability to quick trace calculations • Scenario creation and analysis • Goal-seeking functionality • Graphical user interface
	<p>Integrations:</p> <ul style="list-style-type: none"> • Direct integration with CO\$TAT • Direct application integration with MS-Project • Fully documented API for other application integrations

Upon further review five of the products from the above list were solicited from their respective vendors for a more intensive review. Two products were excluded from additional consideration – PRODUCT B as it was not available under an academic license or for a trial period and PRODUCT D as the product was not made available under the academic license in time for review. After investigating PRODUCT G, it, too, was excluded from further review as it was a model building tool

rather than an algorithmic or data based estimation tool. Furthermore, its primary focus was to provide a structured format for governmental project accounting. One additional tool, PRODUCT E, was eliminated during the research period as it generated errors during several early attempts at use.

An eighth tool, PRODUCT H, was added to the pool for review in an attempt to provide a minimum of three comparators. The preliminary list of features for PRODUCT H is in Table 3 below.

Table 3 – Additional Toolset Preliminary Feature Review

Product	Features
PRODUCT H	Model: <ul style="list-style-type: none"> • Uses simulation with two estimation models (Putnam Model and COCOMO) • Allow for calibration with projects from the user’s organization and project teams or for comparison with industry averages • Create size estimates for multiple modules within a project
	User Experience: Pre-defined project templates or guided wizards to create customized projects
	Integrations: No mention of integrations

5. Research Process

The process used to complete this research were

1. explore the input requirements for each of the three toolsets and determine the suitability of the outputs
2. obtain historical project data for five projects including the original expert estimates and the actual effort required
3. interview the project managers for the respective projects to obtain the required inputs for the three tools
4. obtain estimates from each tool for each project
5. compile and analyze the data

Each of the first four steps is described in section five. Section six contains the summary and analysis of the results.

5.1. Exploration of inputs and outputs for each toolset

PRODUCT C is an Excel add-in that acts as a user interface to calculate a project effort estimate based on the Constructive Cost Model II (COCOMO II). Brief descriptions of the reference models for the various tools are included as an Appendix.

1. Inputs

The product accepts variables required to adjust calculation factors through the use of pop-up forms. Project scope can be entered as expected lines of code or expected function points. There are six project scale factors required which are related to precedentedness, development flexibility, architecture risk/resolution, team cohesion, process maturity, and schedule. The user must rate each of these factors on a six-level scale of Extra High, Very High, High, Nominal, Low, Very Low. Descriptions of each scale level are included in the form. In addition to the project scale factors, the user must rate eleven factors per subsystem related to the product, the platform, the personnel, and the project. Finally, there are an additional five scale factors required for each individual component which are related to the product and the personnel. The user can also enter an average hourly rate which is then multiplied by the total estimated hours to arrive at a cost estimate. A full list of COCOMO II inputs is included as an Appendix.

2. Outputs

PRODUCT C outputs are (1) expected size expressed as thousands of lines of code (KSLOC), (2) expected duration in months, (3) expected effort in person months, (4) average and cumulative staffing, (5) the general productivity rate, and (6) the expected cost. These outputs are in a simple excel spreadsheet. The user can also choose to display a staffing graph which is an option provided from the add-in menu. The projects can be sized as needed by adding subsystems and components via the add-in menu. Multiple estimates can exist in one workbook.

3. Licensing and Distribution

PRODUCT C is offered as a download. There is a \$250 license fee, however, the product is available at no cost for a thirty day trial period.

PRODUCT H is a software tool developed by a company which provides software development training and consulting. PRODUCT H makes use of three estimation approaches -- PRODUCT B, COCOMO II, and Monte Carlo Simulation.

1. Inputs

PRODUCT H can calibrate effort estimates based on one of three approaches:

- a. The least accurate simply compares the project scope against a database of similar projects based on broad definitions of the project type. The project types are generic descriptors such as "Business System" or "Scientific/Engineering System."
- b. The second approach is having the user input a variety of productivity factors similar to the factors used in PRODUCT C.
- c. The third approach is to compare the project against historical projects which were previously estimated by the user for similar scope efforts in the same organization. Scope can be expressed as expected KSLOC, function points, or GUI components.

2. Outputs

PRODUCT H provides a twenty-plus page printable report inclusive of expected (1) staff months, (2) calendar months, (3) costs, (4) peak staff, (5) estimate quality, (6) optimum versus nominal planning data, (7) likelihood of a variety of outcomes based on simulation data, (8) milestones, (9) cashflow analysis, and (10) a summary of cost drivers.

3. Licensing and Distribution

PRODUCT H can be downloaded at www.PRODUCT H.com and is available free of charge under a limited license agreement.

PRODUCT A leverages modeling technology and project-applicable knowledge bases to develop estimates. PRODUCT A includes additional functionality for ongoing project monitoring and management. These functions were not explored for the purposes of this project.

1. Inputs

Product A allows the user to define scope using KSLOC, function points, or use cases. To calibrate the estimate, the user then enters best case, worst case, and most-likely case values for additional project variables, including:

- a. Staffing levels, capabilities & experience
- b. Development environment complexity
- c. Requirements stability
- d. Confidence level
- e. Target/Host systems
- f. Schedule Considerations
- g. Reusability
- h. Integration
- i. Labor rates
- j. Maintenance levels

2. Outputs

PRODUCT A outputs include twenty-seven pre-configured reports and thirty-two pre-configured charts and graphs. In addition, it provides a reporting tool allowing the user to create custom reports via a simple user interface. The preconfigured items include details on all aspects of the estimate including:

- a. Summary of Inputs
- b. Estimated hours
- c. Estimated timeline
- d. Estimated staffing by role
- e. Comparisons to similar projects
- f. Risk analysis
- g. Performance indices
- h. Metrics

3. Licensing and Distribution

PRODUCT A must be obtained via interaction with a sales representative. The product requires purchase of a license and the list price of the product is not known. An evaluation license agreement was completed to obtain a trial copy of the product for this project. The author was offered a complimentary training session by the vendor. The training was not accepted as the author was concerned the training would taint the results of the subjective analysis which includes an Ease of Use assessment.

5.2. Historical project data for research use

Five projects from one organization were chosen for review. The criteria for choosing the projects were primarily pragmatic and included (1) the availability of a baseline expert estimate, (2) the availability of actual effort for the completed project, (3) the availability of the original expert estimator for interview purposes, and (4) an evaluation of the project to ensure scope changes would not have had a material impact on the analysis of the baseline estimate to actual results. A

brief description of each project and a summary of attributes of the chosen projects are presented in the tables below.

Table 4 – Brief Description of Sample Projects

Project	Description
Project 1	Interface for delivery of data from organization’s internal database to an external customer database.
Project 2	Interface for delivery of data from organization’s internal database to an external customer database.
Project 3	Implementation of a COTS imaging and workflow application; included significant business process redesign
Project 4	Implementation of COTS application for workflow and CAPA planning; included development of (1) two custom web forms used as user interfaces for the application and (2) five custom reports
Project 5	Development of custom business intelligence functionality to retrieve data from multiple applications and display data via a custom dashboard; included web interface allowing end user to map data and customized security

Table 5 – Project Attributes

Attribute/Project	Project 1	Project 2	Project 3	Project 4	Project 5
Actual Project length (mos)	6	13	9	4	1.5
Internal v. External Use	External	External	Internal	Internal	Internal
Development Methodology	Waterfall	Waterfall	Waterfall	Waterfall	Agile
Product Type	Developed	Developed	COTS	COTS	Developed
Technology Classification	Established	New Tech	New Tech	New Tech	New Tech

5.3. Project Manager interviews

Three project managers were interviewed to obtain input data required to generate estimates. The project managers were provided with instructions as follows:

1. Attempt to answer all questions based solely on knowledge which was available at the time the baseline expert estimate was created. It is acknowledged that all project managers had the benefit of hindsight at the time of the interviews.
2. Use function points to determine scope rather than lines of code since that variable was known by the managers at the time of interview but was not estimated during the original

baseline exercise. It is acknowledged that none of the project managers possessed expertise in function point analysis. The author assisted with the determination of number of function points.

3. The inputs were obtained by performing a walk-through of each tool and completing the input of data in real-time during the interview process. The data from each project (inputs and outputs) are included in Appendices C, D, and E (removed to protect proprietary information).

5.4. Generate Algorithmic Project Estimates

Inputs obtained during the interview process were reviewed to ensure there were no inconsistencies in the data entered in each toolset. Estimated hours of effort were consolidated into tables for each project which included both the original expert baseline estimate and the actual hours utilized to complete the effort. Each project and tool provided a unique level of detail in terms of hours per role per activity. The PRODUCT C and PRODUCT H tools provided the least amount of detail with a simple total of estimated hours. In order to compare all data equally, evaluations were performed on the totals rather than the detail.

6. Analysis

6.1. Subjective analysis

1. Usability

The International Organization for Standardization (ISO) defines usability as "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" (ISO 9241-11, 1998). Usability consultant Jakob Nielsen expanded the definition to include (Nielsen 2001):

- a. **Learnability:** How easy is it for users to accomplish basic tasks the first time they encounter the design?
- b. **Efficiency:** Once users have learned the design, how quickly can they perform tasks?
- c. **Memorability:** When users return to the design after a period of not using it, how easily can they reestablish proficiency?
- d. **Errors:** How many errors do users make, how severe are these errors, and how easily can they recover from the errors?
- e. **Satisfaction:** How pleasant is it to use the design?

Using the five criteria as defined by Nielsen, the usability of the three products was subjectively rated by the author based on a scale of Excellent, Good, Fair, or Poor. Ratings and comments are included in the tables below.

Table 6 – Summary of Usability Ratings

Criteria/Product	PRODUCT C	PRODUCT H	PRODUCT A
Learnability	Good	Fair	Fair
Efficiency	Excellent	Good	Good
Memorability	Excellent	Fair	Good
Errors	Good	Good	Good
Satisfaction	Good	Fair	Good

Table 7 – Comments on Usability Ratings

Product and Criteria	Rating	Comments
PRODUCT C		
Learnability	Good	Use of the tool seemed to be based on the premise that the user was intimately familiar with the COCOMO II model. Impact of changes to the various scale factors was not easily understood. User had to refer to a model definition manual found on the internet to understand some fields in the tool.
Efficiency	Excellent	The simple interface was easy to use; however, the forms did not display the definitions of the evaluation criteria options until the option was chosen. This design required that the user click through each option in order to evaluate the range of choices.
Memorability	Excellent	User interface was a single MS Excel worksheet with an Add-in menu. Once the interface was used once, it was extremely simple to use again. There were limited menu options thereby increasing the ability to memorize available functionality.
Errors	Good	No errors in calculation were noted. Input errors were easily overwritten.
Satisfaction	Good	User satisfaction was limited to fair due to the limited amount of output provided based on the time required to enter all input variables.
PRODUCT H		
Learnability	Fair	Menus were not arranged in obvious order of operations. Some items in the panel were actually buttons, although that fact was not apparent until the user scrolled over the area.
Efficiency	Good	The interface was limited in the amount of required inputs and, once the user understood the logical order of operations, it was fairly quick to create an estimate.
Memorability	Fair	User had some challenges when stepping through necessary steps to create an estimate. Order of operations to chose calibration type was not immediately obvious, particularly when not using the wizard.
Errors	Good	No errors in calculation were noted. Input errors were easily overwritten.
Satisfaction	Fair	User satisfaction was limited to fair based on the user's frustration that estimates were noticeably different than other estimates thereby requiring additional reviews to assess variations. Additionally, interface was not visually appealing. Finally, the tool states that the most reliable estimates require user entry of historical projects from their organization.
PRODUCT A		
Learnability	Fair	The tool required substantially more input than the other toolsets tested. As such, it took some time and heavy use of the Help functionality to successfully complete the first estimate.
Efficiency	Good	In spite of the required volume of data, the tool felt efficient as the user could step through all the required entry using a series of "show details" type menus. This approach allowed the user to gauge progress throughout the usage period.
Memorability	Good	Although the first use of the product took longer than use of the other toolsets, the user was easily able to move through the required steps in subsequent uses.
Errors	Good	No errors in calculation were noted. Input errors were easily overwritten.

Product and Criteria	Rating	Comments
Satisfaction	Good	User was extremely satisfied with the experience. Although the product required many more inputs, the depth and breadth of available outputs made the process seem more useful and therefore more satisfying. Furthermore, the wide variety of options prevented the user from experiencing the feeling of 'fitting a square peg into a round hole.' Additionally, the interface was visually appealing and the multiple panes made changing between input and output views very intuitive.

In addition to the usability ratings, the author feels strongly that two additional measures related to functionality should also be considered as part of the overall user experience. One is Flexibility and the other is Breadth of Functionality. These factors are being included because, even though PRODUCT C was rated highest in the categories above, the user had a stronger feeling of satisfaction when using the PRODUCT A tool. That satisfaction can be attributed to PRODUCT A's wider range of options for both fine-tuning the estimate and for including data related to a broader range of project types, related to ongoing maintenance, and related to estimated versus actual project performance. A rating of Excellent, Good, Fair, or Poor is provided for the two additional categories below.

Table 8 – Summary of Additional Usability Ratings

Criteria/Product	PRODUCT C	PRODUCT H	PRODUCT A
Flexibility	Fair	Fair	Excellent
Breadth of Functionality	Poor	Good	Excellent

2. Level of difficulty required to obtain input data

Each tool's required set of input variables are very similar in nature, however, there are some differences in the knowledge level required.

PRODUCT C requires input variables based solely on the COCOMO II model. The variables are primarily subjective in nature and require the opinion of an expert in the organization. The objective inputs must be either expected lines of code or expected function points. The author's

opinion is that PRODUCT C requires an nominal level of effort to obtain input data but a high level of expertise with both the organization and the product in order to create a meaningful estimate.

PRODUCT H offers three types of estimates. The first type is based solely on comparison of the project type to projects of similar types. The level of effort and expertise to obtain an estimate of this nature is low, however, the tool will warn the user that the quality of the estimate is low as well. The second type is based on project attributes and constraints. The level of effort and expertise to obtain this type of estimate is rated as nominal and is very similar in nature to the effort required for the PRODUCT C tool. The third type of estimate is based on comparison of the project to projects in the same organization. The effort required to obtain this type of estimate is rated as high due to the fact the tool must be used over a reasonably long period of time in order to gather enough data to provide meaningful estimates.

PRODUCT A requires the most effort and expertise in both the organization and the product. The inputs are similar to the COCOMO II inputs, however, there are several other layers of details that relate to maintenance, integrations, multiple modules, multiple development options, and metrics. The input effort required for PRODUCT A is rated as high, however, it is noted that the good usability of the tool offset some of the perceived difficulty in obtaining the detailed inputs.

Table 9 – Summary of Effort Level for Inputs

Product	Required Expertise	Required Effort
PRODUCT C	Nominal	Nominal
PRODUCT H	Low to Nominal	Low to Nominal
PRODUCT A	High	High

3. Set-up and maintenance overhead required

The versions of each product utilized for this research were single user. As such installation and setup effort was minimal. The simplest of the three was the installation of the Excel add-in

required for PRODUCT C. However, it is noted that the product requires the user to have previous ownership of Microsoft Excel. Both PRODUCT H and PRODUCT A installed easily and quickly and there is no noted difference in the required set-up and maintenance of the three products. PRODUCT A does offer an enterprise edition allowing for the collection of all organizational data in a shared environment. As this product was not available under the granted license, the author can not offer an opinion on the ease or difficulty of that version of the product.

6.2. Objective analysis

Results for each project estimate are included below in Tables 9 - 18. The initial table shows the results based on a comparison between the estimates and all recorded actual. Upon review, the author made a series of adjustments to the actual to remove the following actual hours:

- a. Training hours (if applicable) were removed as none of the tools indicated that training or change management efforts were included in the estimation algorithms.
- b. Hours charged to the project by a subject matter expert (SME). Although the hours spent by the SMEs were often charged to requirement gathering or testing activities, the assumption by the author is that the estimation tools, which are designed from the perspective of a product provider, would not include hours spent by SMEs on business process redesign and user acceptance testing.

1. Project 1 Data Review

Estimated values provided by the expert were close in value to both the PRODUCT C and PRODUCT A estimates. The PRODUCT H was notably lower, however, as will be revealed from the remaining data reviews, the PRODUCT Hs were either significantly lower for small projects or significantly higher for large projects. Therefore, for the purposes of the narrative data reviews, speculation about the inconsistency of the PRODUCT H will be avoided. Upon investigation of the unique characteristics of project 1, it was noted that project 1 was very similar in nature to Project 2 which was actually executed prior to Project 1. It is the belief of the author that familiarity with the platform and product can provide a more significant scaling factor than is evidenced by the algorithms.

Table 10 – Project 1 Effort Estimates versus Actuals; no adjustments

Task	Actual Hours	Expert Estimate Hours	PRODUCT		
			PRODUCT C Hours	A Hours	PRODUCT H Hours
Deployment	12	9		35	
Design and Development	261.5	473.5	954.22	600	230.2
Project Management	145.75	128		101	
Qualification	40	59			
Requirements	63.5	28.5		104	
Support		4			
Testing	185.5	251		234	
Training	2.5	22			
Total	710.75	975	954.22	1074	230.2

Table 11 – Project 1 Effort Estimates versus Actuals; adjusted for training and SME hours

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Deployment	7	9		35	
Design and Development	202.5	336.5	954.22	600	230.2
Project Management	139.25	116		101	
Qualification	40	59			
Requirements	37.75	24.5		104	
Testing	178.5	251		234	
Total	605	796	954.22	1074	230.2

2. Project 2 Data Review

Project 2 was undertaken by a project team with only basic training in the platform and product. In complete agreement with the results from Project 1, Project 2 reveals that a project in which the development team is unfamiliar with the platform, product, and language can have unexpectedly high overages in delivery hours. Although the expert, PRODUCT C, and PRODUCT A estimate are reasonably close in value, the actuals are again significantly different indicating that the factors related to familiarity may not carry enough weight in the calculations.

Table 12 – Project 2 Effort Estimates versus Actuals; no adjustments

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Deployment	432.5	40.25		112	
Development	4751	1108.75	2609.98	2188	503
Requirements	628	483.25		323	
Testing	5325.25	2451		741	
Grand Total	11,136.75	4083.25	2609.98	3364	503

Table 13 – Project 2 Effort Estimates versus Actuals; adjusted for training and SME hours

Task	Actual Hours	Expert Estimate Hours	PRODUCT		
			PRODUCT C Hours	A Hours	PRODUCT H Hours
Deployment	432.5	14		112	
Development	4751	1083.75	2609.98	2188	503
Requirements	628	459.25		323	
Testing	4475	1603		741	
Grand Total	10,286.5	3160	2609.98	3364	503

3. Project 3 Data Review

Data for Project 3 is difficult to interpret. The project was the configuration and implementation of a COTS product to execute imaging and workflow management. A project of this nature was not previously executed by the particular team who provided the estimation. Furthermore, the estimation was not based on a work breakdown structure (WBS) but instead based on an estimate of the percentage of particular resources that would be required to execute the project. As is evidenced by the vast discrepancy between the expert estimate and the actual hours, it is clear that the resources either were not required to be utilized at the anticipated levels or were consumed by other responsibilities during the project. Discussion with the project manager indicates that the resources were not required at the anticipated levels. The algorithmic estimates provided by the tools were notably lower than the actual results. However, it can be noted that the PRODUCT A hours, which have task level detail, indicate the lower hours are primarily in the areas of requirements and project management. Discussion with the project manager revealed that this particular project's requirements gathering sessions were heavily weighted with business process redesign of multiple global processes. As there is not a variable to address business process redesign in the algorithms, it is possible that this variable can account for the significant difference in requirements and management.

Table 14 – Project 3 Effort Estimates versus Actuals; no adjustments

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Deployment	193	360		57	
Development Project	458.5	1440	1387	856	305.8
Management	793	1440		144	
Requirements	1429.25	2592		130	
Testing	632.75	3960		388	
Grand Total	3506.5	9792	1387	1575	305.8

Table 15 – Project 3 Effort Estimates versus Actuals; adjusted for training and SME hours

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Deployment	85	360		57	
Development Project	458.5	1440	1387	856	305.8
Management	793	1440		144	
Requirements	741	1440		130	
Testing	368.75	2520		388	
Total	2446.25	7200	1387	1575	305.8

4. Project 4 Data Review

Project 4 was very similar in nature to Project 3 as the requirements sessions were heavily weighted towards redesign and globalization of business processes. Additionally, it was more difficult to use PRODUCT H and PRODUCT C for these COTS projects as there were not options for clearly delineating the efforts for development from those required for configuration. The author also has concerns that testing hours were not accurately entered into the system of record. Discussions with the project manager clearly indicated that testing was of particular importance to this business team.

Table 16 – Project 4 Effort Estimates versus Actuals; no adjustments

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT	
				A Hours	PRODUCT H Hours
Development Project	631	437.5	12,033	417	2080.58
Management	574.5	733		67	
Testing	27	0		273	
Requirements	1991.25	3504		47	
Total	3223.75	4674.5	12,033	804	2080.58

Table 17 – Project 4 Effort Estimates versus Actuals; adjusted for training and SME hours

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT	
				A Hours	PRODUCT H Hours
Development Project	631	437.5	12033	417	2080.58
Management	574.5	733		67	
QA	27	0		273	
Requirements	390.5	614.5		47	
Total	1623	1785	12,033	804	2080.58

5. Project 5 Data Review

Project 5 was, by far, the simplest of all the observed projects. It was primarily executed by one IT professional who functioned as project manager, business analyst, designer, and developer. Testing was performed by an independent party, however, it is noted that the hours spent on testing were not accurately captured in the system of record.

Table 18 – Project 5 Effort Estimates versus Actuals; no adjustments

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Management				20	
Requirements		40		13	
Testing		40		63	
Total	275	200	122.4	225	34.38

Table 19 – Project 5 Effort Estimates versus Actuals; adjusted for training and SME hours

Task	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours
Development	275	120	122.4	129	34.38
Project Management				20	
Requirements		40		13	
Testing		40		63	
Total	275	200	122.4	225	34.38

6. Consolidated Data Review

When reviewing the total project hours with no specification related to specific task, the PRODUCT C tool appears to be the most accurate of the toolsets tested both with and without the actual hours being adjusted for training and SME hours. However, it is impossible to determine the validity of the PRODUCT C model at a level that would be useful enough for project planning purposes in the absence of hours by task or role.

When reviewing the more detailed task-based hours provided by the PRODUCT A tool, it is notable that the PRODUCT A estimates are less accurate than the expert estimates in all categories. When reviewing the comparisons of data based on project attribute, there is only one category where a tool appeared to be more accurate than the expert estimate. In the COTS category, the PRODUCT A tool estimate was significantly better than the expert estimate.

Table 20 – Project Totals in Hours; with and without adjustments for training and SME hours

Project w/o adj	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert % Diff from Actual	SCE % Diff from Actual	PRODUCT A % Diff from Actual	Cons % Diff from Actual
Project 1	710.75	975	954.22	1074	230.2	31.57%	57.72%	77.52%	-61.95%
Project 2	11136.75	4083.25	2609.98	3364	502.52	-69.28%	-74.63%	-67.30%	-95.11%
Project 3	3506.5	9792	1387	1575	305.8	194.33%	-43.30%	-35.62%	-87.50%
Project 4	3223.75	4674.5	12033	804	2080.58	9.98%	641.40%	-50.46%	28.19%
Project 5	275	200	122.4	225	34.38	-27.27%	-55.49%	-18.18%	-87.50%
Total	18852.75	19724.75	17106.6	7042	3153.48	-13.75%	12.28%	-53.78%	-79.30%

Project w/ adj	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert % Diff from Actual	SCE % Diff from Actual	PRODUCT A % Diff from Actual	Cons % Diff from Actual
Project 1	605	796	954.22	1074	230.2	37.18%	34.26%	51.11%	-67.61%
Project 2	10286.5	3160	2609.98	3364	502.52	-63.34%	-76.56%	-69.79%	-95.49%
Project 3	2446.25	7200	1387	1575	305.8	179.25%	-60.44%	-55.08%	-91.28%
Project 4	1623	1785	12033	804	2080.58	45.00%	273.26%	-75.06%	-35.46%
Project 5	275	200	122.4	225	34.38	-27.27%	-55.49%	-18.18%	-87.50%
Total	15235.75	13141	17106.6	7042	3153.48	4.63%	-9.26%	-62.65%	-83.27%

Table 21 – Design and Development Totals in Hours

Project	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	PRODUCT A Diff from Actual
Project 1	261.5	473.5	954.22	600	230.2	212	338.5
Project 2	4751	1108.75	2609.98	2188	502.52	3642.25	-2563
Project 3	458.5	1440	1387	856	305.8	981.5	397.5
Project 4	631	437.5	12033	417	2080.58	-193.5	-214
Project 5	275	120	122.4	129	34.38	-155	-146
Total	6377	3579.75	17106.6	4190	3153.48	2797.25	-2187

Table 22 – Project Management Totals in Hours

Project	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	PRODUCT A Diff from Actual
Project 1	145.75	128		101		-17.75	-44.75
Project 3	793	1440		144		647	-649
Project 4	574.5	733		67		158.5	-507.5
Project 5				20		0	20
Total	1513.25	2301		332		787.75	-1181.25

Table 23 – Requirements Totals in Hours

Project	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	PRODUCT A Diff from Actual
Project 1	63.5	28.5		104		-35	40.5
Project 2	628	483.25		323		-144.75	-305
Project 3	1429.25	2592		130		1162.75	-1299.25
Project 4	1991.25	3504		47		1512.75	-1944.25
Project 5		40		13		40	13
Total	4112	6647.75		617		2535.75	-3495

Table 24 – Deployment Totals in Hours

Project	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	PRODUCT A Diff from Actual
Project 1	12	9		35		-3	23
Project 2	432.5	40.25		112		-392.25	-320.5
Project 3	193	360		57		167	-136
Grand Total	637.5	409.25		204		-228.25	-433.5

Table 25 – Testing Totals in Hours

Project	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	PRODUCT A Diff from Actual
Project 1	225.5	310		234		84.5	8.5
Project 2	5325.25	2451		741		2874.25	-4584.25
Project 3	632.75	3960		388		3327.25	-244.75
Project 4	27	0		273		-27	246
Project 5		40		63		40	63
Grand Total	6210.5	6761		1699		550.5	-4511.5

Table 26 –Totals in Hours by Project Type

Project Type	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	SCE Diff from Actual	PRODUCT A Diff from Actual	Cons Diff from Actual
COTS	6730.25	14466.5	13420	2379	2386.38	7736.25	6689.75	-4351.25	-4343.87
Dev	12122.5	5258.25	3686.6	4663	767.1	-6864.25	-8435.9	-7459.5	-11355.4
Total	18852.75	19724.75	17106.6	7042	3153.48	872	-1746.15	11810.75	-15699.3

Table 27 –Totals in Hours by Project Length in Months

Project length in months	Sum of Actual	Sum of Expert Estimate	Sum of PRODUCT C	Sum of PRODUCT A	Sum of PRODUCT H	Expert Diff from Actual	SCE Diff from Actual	PRODUCT A Diff	Cons Diff
1.25	275	200	122.4	225	34.38	-75	-152.6	-50	-240.62
4	3223.75	4674.5	12033	804	2080.58	1450.75	8809.25	-2419.75	-1143.17
6	710.75	975	954.22	1074	230.2	264.25	243.47	363.25	-480.55
9	3506.5	9792	1387	1575	305.8	6285.5	-2119.5	-1931.5	-3200.7
13	11136.75	4083.25	2609.98	3364	502.52	-7053.5	-8526.77	-7772.75	-10634.2
Total	18852.75	19724.75	17106.6	7042	3153.48	872	-1746.15	11810.75	-15699.3

Table 28 –Totals in Hours by Project designed for Internal or External Use

Internal or External	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	SCE Diff from Actual	PRODUCT A Diff from Actual	Cons Diff from Actual
External	11847.5	5058.25	3564.2	4438	732.72	-6789.25	-8283.3	-7409.5	-11114.8
Internal	7005.25	14666.5	13542.4	2604	2420.76	7661.25	6537.15	-4401.25	-4584.49
Total	18852.75	19724.75	17106.6	7042	3153.48	872	-1746.15	11810.75	-15699.3

Table 29 –Totals in Hours by Development Methodology

Dev Methodology	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	SCE Diff from Actual	PRODUCT A Diff from Actual	Cons Diff from Actual
Waterfall	18577.75	19524.75	16984.2	6817	3119.1	947	-1593.55	11760.75	-15458.7
Agile	275	200	122.4	225	34.38	-75	-152.6	-50	-240.62
Total	18852.75	19724.75	17106.6	7042	3153.48	872	-1746.15	11810.75	-15699.3

Table 30 –Totals in Hours by Technical Classification

Tech Classification	Actual Hours	Expert Estimate Hours	PRODUCT C Hours	PRODUCT A Hours	PRODUCT H Hours	Expert Diff from Actual	SCE Diff from Actual	PRODUCT A Diff from Actual	Cons Diff from Actual
Established	710.75	975	954.22	1074	230.2	264.25	243.47	363.25	-480.55
New Tech	18142	18749.75	16152.38	5968	2923.28	607.75	-1989.62	-12174	-15218.7
total Total	18852.75	19724.75	17106.6	7042	3153.48	872	-1746.15	11810.75	-15699.3

7. Summary of Conclusions and Recommendations

7.1. Summary of Conclusions

The author has drawn the following conclusions from the data presented in preceding sections:

1. Based on subjective criteria, the toolsets reviewed were ranked from best to worst in the order of PRODUCT A, PRODUCT C, PRODUCT H.
2. Based on the objective criteria, PRODUCT C was noted to be the most accurate tool in providing total estimated hours.
3. Based on the objective criteria, expert estimates were noted to be more accurate than any of the reviewed toolsets.

7.2. Recommendations for Future Research

There are several shortcomings noted throughout the research process. In order to promote more meaningful results in future research on this topic, the following recommendations are offered:

1. Increase the sample size

The original plan was to use a selection of fifteen sets of actual project data and five toolsets. Many of the datasets were rejected for research use after initial review. They were deemed unsuitable because either the expert estimation or the actual effort was not captured at a level of detail to allow for analysis. In order to collect a more significant sample of data, the author would recommend reviewing three times the number of projects that are desired for analysis.

2. Expand the subjective analysis

The author was the sole user of the toolsets during this research process. The expert estimators were exposed to the toolsets during live walk-throughs of the toolsets to capture the required inputs. A subjective analysis inclusive of multiple viewpoints would be more significant than the use of a single viewpoint. If the expert estimators could be trained to input the data themselves, their usability ratings could be captured as well.

3. Select projects of varying lengths

All projects included in the research were short in duration. Although several longer projects were reviewed for use, they were discarded due to the lack of detail available in project estimates and actuals for older projects. The analysis of projects of diverse length may have provided different results as at least one of the toolsets reviewed specifically disclaimed accuracy for projects of short duration.

4. Select projects from multiple organizations and multiple estimators

All projects reviewed for this research were obtained from a single organization. Furthermore, three of the project were estimated and managed by a single project manager. Because the projects were from one organization and/or one manager, they may not be representative of an average project. They could be subject to particularly poor or good practices which could skew the validity of the results.

5. Provide training on estimation techniques

It is the author's opinion that delineating between "expert estimation" and "formal methodology" is somewhat misleading. In practice, the input of data into a toolset requires a similar, if not higher, level of expertise. The project managers who participated in the research expressed more comfort in providing an expert estimate than in determining specific input values for use with the tools. For example, it was deemed easier to state that it would take a developer three weeks to code three reports than it was to review the requirements and determine the functionality required of each report and translate that functionality into input variables. In order to get results from the tools that were created with a reasonable level of expertise, the estimators should be trained in the estimation methodology required by the tools (e.g. KSLOC or function point estimation).

7.3. Recommendations for an Improved Project Effort Estimation Tool

1. References to formal models should be defined within the tool

Any reference to a model or a variable should have a definition included within the tool. Tools which assume user knowledge or direct the user to other sources do not rate well for usability. Furthermore, rating scales should contain clearly visible explanations of what they indicate. For example, a rating of "nominal" should include examples of what a nominal result would be. These examples could be in the form of a mouse-over or a drop-down.

2. Menus should provide a walk-through of logical steps

Although wizards are useful for a novice user, the menu format should also reflect a series of logical steps so that it becomes intuitive for a user to step through detailed set up processes or edits of existing estimates.

3. Criteria should allow for user-defined items

Although the standard algorithms may suffice for many organizations or project types, some specific projects may require the addition of specific factors that are known to affect the overall effort. Allowing the user to define and weight those factors could be useful and allow an organization to refine the algorithms to meet specific needs unique to their environment.

4. Detail should be available at both a macro and micro level

Two of the toolsets provided only macro-level summaries of effort hours. These tools proved much less useful when attempting to compare the estimates to actual results and would be virtually useless for creating a baseline project plan.

5. Analysis tools should be expanded

Since estimation tools are primarily used to assess the viability or cost of a potential project, it would be useful to expand the analysis tools to accommodate the capture of non-labor costs and the value over time of the finished product. With the combination of fully loaded costs and return, an automated ROI calculation could be included in the output.

8. Works Cited

Blattberg, R., & Hoch, S. (1990). Database Models and Managerial Intuition: 50% Model + 50% Manager. *Management Science*. August 1990. p. 887.

Boehm, B., Abts, C. & Chulani, S. (2000). Software Development Cost Estimation Approaches – A Survey. *Annals of Software Engineering*. Volume 10. Issue 1-4.

Chemuturi, M. (2009). Analogy based Software Estimation. Published as a whitepaper on www.chemuturi.com.

Cuadrado-Gallegoa, J., Rodriguez-Soriaa, P., & Martin-Herreraa, B. (2010). Analogies and differences between Machine Learning and Expert based Software Project Effort Estimation. *Software Engineering Artificial Intelligence Networking and Parallel/Distributed Computing (SNPD), 2010 11th ACIS International Conference*. June 2010. pp. 269-276.

ISO 9421-11 (1998). Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability. www.iso.org.

Jorgensen, M. (2004). A Review of Studies on Expert Estimation of Software Development Effort. *J. Systems and Software*. Volume 70, nos. 1–2. pp. 37–60.

Jorgensen, M. (2007). Estimation of Software Development Work Effort: Evidence on Expert Judgment and Formal Models. *International Journal of Forecasting*. 23(3):449-462.

Jorgensen, M., Boehm, B., & Rifkin S. (2009). Software Development Effort Estimation: Formal Models or Expert Judgment? *IEEE Software*. March 2009. 14-19.

Jorgensen, M. & Molokken, K. (2003). A Review of Surveys on Software Effort Estimation. *2003 International Symposium on Empirical Software Engineering*. September 2003. p. 223.

Jorgensen, M., & Molokken-Ostvold, K. (2004). Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method. *IEEE Transactions on Software Engineering*. December 2004. pp. 993-1007.

Jorgensen, M., & Shepperd, M. (2007) A Systematic Review of Software Development Cost Estimation Studies. *IEEE Transactions on Software Engineering*. 33(1):33-53

Meli, R., & Santillo, L. (1999). Function point estimation methods: a comparative overview. FESMA 98 – The European Software Measurement Conference – Amsterdam, October 6-8.

Nielson, J. (2001) First Rule of Usability? Don't Listen to Users. Jakob Nielsen's Alertbox, Use IT website. ISSN 1548-5552. August 5th 2001.

Appendix A – Estimation Models Referenced in PRODUCT H Documentation

PRODUCT B

Removed to protect proprietary information.

Cocomo II

Cocomo II is a continuation of the work begun by Barry W. Boehm in the 1970s and described in his 1981 book, *Software Engineering Economics* (Prentice-Hall). Since 1981, additional work has been done to refine the Cocomo model and adapt it to projects other than the U.S. Department of Defense projects for which it was originally developed. At present, the model has been extended into Cocomo II which allows estimates to be created for virtually any kind of project by specifying a set of cost drivers.

Monte Carlo Simulation

Monte Carlo simulations use repeated sampling to determine the properties of some phenomenon (or behavior). They can be used to model complex interactions in the face of uncertain estimating assumptions.

Appendix B – Planned Capstone Project Timeline versus Actual Capstone Project Timeline

As shown in the timelines below, the original estimate for obtaining toolsets and data took significantly longer than anticipated. It is recommended for future projects that steps which are dependent on the cooperation of third parties should have a significant time buffer. Additionally, the time required to learn how to use the toolsets was not reflected in the original estimate. At least two weeks should have been included for toolset practice and review. In total the project was almost two months behind schedule and the allotment of time for analysis was three weeks shorter than originally anticipated.

Figure 1 – Planned versus Actual Capstone Project Timeline

Task Name	Proposed			Actual		
	Duration in Days	Start	Finish	Duration in Days	Start	Finish
Obtain proposal approval	1	Mon 20/06/11	Mon 20/06/11	1	Tue 28/06/11	Tue 28/06/11
Edit proposal based on committee recommendation	7	Mon 20/06/11	Mon 27/06/11			
Obtain 3 toolsets for further research				105	Tue 05/07/11	Mon 17/10/11
Complete expanded feature set comparison	14	Mon 27/06/11	Mon 11/07/11	44	Thu 01/09/11	Fri 14/10/11
Choose 3 toolsets for further research	2	Mon 11/07/11	Wed 13/07/11			
Edit proposal based on committee recommendation				7	Mon 17/10/11	Sun 23/10/11
Obtain input and output data for historical projects	21	Wed 13/07/11	Wed 03/08/11	40	Thu 15/09/11	Mon 24/10/11
Run data through estimation tools	14	Wed 03/08/11	Wed 17/08/11	7	Tue 25/10/11	Mon 31/10/11
Analyze and Document Results	26	Wed 17/08/11	Mon 12/09/11	10	Wed 02/11/11	Fri 11/11/11
Submit Rough Draft to Advisor				0.5	Fri 11/11/11	Fri 11/11/11
Edit based on advisor recommendation and document conclusions	0.5	Mon 12/09/11	Mon 12/09/11	5	Sat 12/11/11	Wed 16/11/11
Finalize Capstone Paper	21	Tue 13/09/11	Mon 03/10/11	1	Thu 17/11/11	Thu 17/11/11
Capstone Presentation	1	Tue 04/10/11	Tue 04/10/11	1	Tue 29/11/11	Tue 29/11/11
Total	107.5			154		

Appendices C, D, and E removed to protect proprietary information