Using Value-Based Total Cost of Ownership (TCO) Measures to Inform Subsystem Trade-offs

Nicole M. Radziwill*[a,b], Ronald F. DuPlain[b]
[a]James Madison Univ., Dept. of Integrated Science & Technology, Harrisonburg, VA USA 22807;
[b]National Radio Astronomy Observatory, 520 Edgemont Rd., Charlottesville VA USA 22902

ABSTRACT

Total Cost of Ownership (TCO) is a metric from management accounting that helps expose both the direct and indirect costs of a business decision. However, TCO can sometimes be too simplistic for "make vs. buy" decisions (or even choosing between competing design alternatives) when value and extensibility are more critical than total cost. A three-dimensional value-based TCO, which was developed to clarify product decisions for an observatory prior to Final Design Review (FDR), will be presented in this session. This value-based approach incorporates priority of requirements, satisfiability of requirements, and cost, and can be easily applied in any environment.

Keywords: metrics, cost analysis, Total Cost of Ownership, TCO, lifecycle costing

1. INTRODUCTION

There is often a significant difference between the *price* of something at the time of purchase, and the *cost* of that thing over its useful life (and even beyond, if the costs of disposal or decommissioning are a concern). As a result, it is often important for observatories and funding agencies to make strategic investment decisions on the basis of *lifecycle cost* rather than the start-up price of building and commissioning an instrument alone. Total Cost of Ownership (TCO) is one metric that has been devised to express the full lifecycle cost of a project or venture. Unfortunately, lifecycle costs do not typically capture important considerations such as availability of skill sets to maintain a system, or how easily a system will be to extend and continually improve.

Sometimes, the lowest cost solution is not the best selection The purpose of this paper is to present the theoretical and practical foundations for a specific *value-based* Total Cost of Ownership (TCO) metric called "3D TCO". Furthermore, a case study will describe how the technique was successfully used as one observatory selected the software components for its cyberinfrastructure and defended the decision to its funding agency.

According to the Random House dictionary [1] value is defined as "relative worth, merit or importance" (Figure 1). Whereas TCO deals with costs only, a value-based measure would explore such costs *relative to other factors* or even relative to other potential solution candidates. These other factors could include levels of quality, staff familiarity with a product or a technology, or agility of the solution. Analytically, this implies that in contrast with the dollar figure that represents the TCO for a possible solution, a value-based TCO would be a chart illustrating two or more dimensions.

> **val·ue**
>
> [**val**-yoo] Show IPA ***noun, verb,*** **-ued, -u·ing.**
>
> ***–noun***
>
> **1.** relative worth, merit, or importance: *the value of a college education; the value of a queen in chess.*

Figure 1. Value as defined by the Random House dictionary.

*nicole.radziwill@gmail.com; Phone +1 703-835-6336; Fax +1 703-842-8074

2. THE TOTAL COST OF OWNERSHIP (TCO) CONCEPT

Total Cost of Ownership (TCO) is one approach to lifecycle costing for a potential or planned strategic investment. The method provides one option for exploring strategic cost management, which includes consideration of three factors: value chain analysis, strategic positioning analysis, and cost driver analysis. The TCO approach is used for only one of these aspects of cost management, value chain analysis, because calculating TCO requires estimating all of the activities needed to bring a product or service to its user community and adding all the elements over an appropriate time horizon.

2.1 Definition and Time Horizon

The concept of TCO, and in particular of using metrics to objectively evaluate purchasing decisions, has been applied since the 1920's. [2] Throughout its history, researchers and practitioners have identified many possible cost elements that may be included in a TCO calculation. Table 1 lists examples of cost elements that could potentially be included in

Possible Components in a Total Cost of Ownership (TCO) Calculation	
[3] Ellram (1995)	Price paid for item, cost of order placement, cost of research and qualification of suppliers; procurement costs (e.g. transportation), receiving, inspection, rejection, replacement, downtime, disposal
[4] Ellram & Siferd (1998)	A case study presented by these authors includes the following components of TCO: price per unit, corporate contracting, divisional purchasing, materials engineering, transportation and logistics, receiving, inspection and screening costs, storage, delivery, waste disposal, pre-sale and post-sale quality assurance
[5] David et al. (2002)	Acquisition of hardware and software; control costs of centralization and standardization; operations costs including support, evaluation, installations and upgrades, training, downtime, personal use of company resources, auditing, virus protection, power consumption
[6] Bhutta & Huq (2002)	Manufacturing costs such as raw materials, labor and depreciation of machines; quality costs such as inspection, rework, and costs due to delays in shipping, receiving, or processing; technology costs such as designing and engineering; after-sales service costs
[7] BEA, Inc. (2003)	Server, storage, and infrastructure costs (and upgrades); operating system and applications costs, including licensing; ongoing software and hardware maintenance fees; end user, developer and admin education; installation, configuration, integration and development support; ongoing costs for space and electricity; opportunity costs such as time delays resulting in revenue and/or productivity loss
[8] Castellani et al. (2005)	Hardware, software, consumables, cost of network management and administration, infrastructure costs including floor space for devices and supplies, maintenance and support costs, loss of productivity due to inadequate, inappropriate, or poorly functioning equipment
[9] Jones (2007)	Initial cost of building an application, cost of enhancing the application with new features, cost of repairing defects and bugs, cost of customer support, cost of refactoring, removal of error-prone modules

Table 1. Examples of elements that may be included in TCO calculations.

TCO calculations. The time horizon over which to sum cost elements is also significant in TCO calculations, and may be 3 years, 5 years, 10 years, or another meaningful frame of reference. For example, the construction period for a telescope may be the appropriate time frame, or the time over which the funding environment can be expected to be stable. It is important that all TCO calculations that will be compared with one another are calculated over the same time frame.

2.2 Utility of TCO

Using TCO, an organization can compare different investment alternatives based on their long-term impact on the organization. Specific questions that can be answered when considering strategic software and cyberinfrastructure investment decisions include the following:

- Should our organization purchase Commercial Off-the-Shelf (COTS) software and customize it?
- Should our organization develop its own software subsystems from scratch?
- Should we use open source software, recognizing that the lifecycle cost of open source software can exceed the initial installation costs by several orders of magnitude? [10]
- Should we consider Software as a Service (SaaS) even though it requires relinquishing some control over our applications?

These trade-offs are evaluated by calculating TCO for each candidate solution and comparing them to one another. However, straight comparison of TCO calculations may lead an organization to choose the lowest cost solution when a more expensive solution would actually provide greater long-term value. It is for this reason that value-based TCOs can be more effective for strategic investment decisions than currency-based TCOs.

2.3 Currency-Based vs. Value-Based TCOs

There are two categorical approaches to TCO calculation that have been utilized over the past few decades: currency-based and value-based. The currency-based approach is the most straightforward, and requires collecting actual cost data for estimates for each of the TCO elements. The value-based approach seeks to integrate qualitative considerations into the TCO calculation, often by assigning point values to key decision criteria such as quality, delivery, technology and support. Each TCO element is weighted by the decision criteria to create a composite TCO that embeds the qualitative factors.

However, as noted by Ellram [3], “value-based models require a great deal of fine tuning and effort to develop the proper weightings and point allocations so that they reflect the TCO.” The choice of weightings is necessarily subjective, and may change over time as an organization’s priorities change. As a result, most of the value-based approaches to calculating TCO are limited by these subjective considerations, reducing their utility.

2.4 Barriers and Limitations of TCO Analysis

The TCO approach is not without limitations. Ellram [9] notes that there are many challenges that can make implementing TCO in practice difficult, and at times, even impossible. These include:

- Lack of relevant data, or inability to acquire data
- Difficulty of determining the appropriate cost elements to include in the TCO calculation
- Complexity of TCO calculation itself
- Organizational culture is resistant to characterizing investment decisions using TCO or using the metric for strategic cost management
- Irrelevance of analysis in certain situations

Alternatives may require complex modeling, however, and TCO remains a popular approach to evaluate trade-offs.

3. CASE STUDY: 3D TCO FOR SOFTWARE PRODUCT SELECTION

3.1 Overview

The value-based 3D TCO method was developed to help a new observatory select four of its software subsystems: 1) a database, 2) a messaging system to broker inter-application communication, 3) a data processing system to manage its data processing workflows/automated pipeline, and 4) an asset management system for version control and configuration management of software systems and data collection hardware (i.e., the equivalent of its receivers and backends).

The observatory (whose identity is concealed as a result of NDA) was interested in determining two things: first, which COTS software products to purchase for each of its subsystems, and second, whether it would better to purchase those products or build them from scratch. The initial investigations into these questions, which were focused on qualitative considerations (e.g. quality, perceived extensibility, developer skill sets/comfort levels), indicated that the cheapest solutions would not provide the greatest value. Because the observatory and its funding agency was seeking a more quantitative way to illustrate their concerns to review panels, the concept of 3D TCO was conceived.

3.2 3D TCO Charts

The solution to this challenge was to display currency-based TCO such that value-based considerations could be more

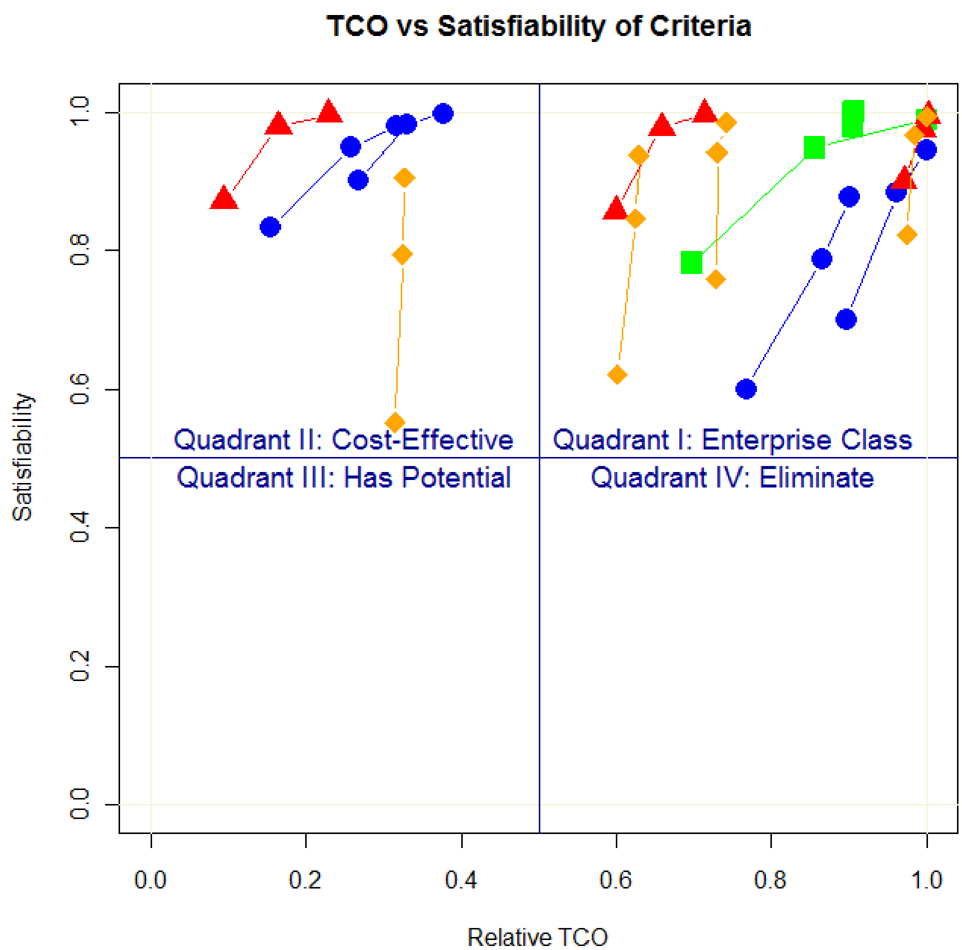

Figure 2. 3D TCO for four potential software subsystems: asset management (triangles), messaging (circles), data processing (diamonds), and databases (squares).

easily communicated to an audience which included managers and the funding agency. To do this, relative satisfiability (the proportion of requirements that could be satisfied at each priority, expressed as a percentage of the total requirements that could be satisfied by the most comprehensive solution) was plotted against relative TCO (the TCO for each product at each priority level, expressed as a percentage of the highest possible TCO). Figure 1 shows the raw data.

- **Quadrant I** represents the enterprise-class solutions. These options typically satisfy a breadth of the project's requirements, but are also comparatively very expensive.
- **Quadrant II** contains the solutions that are relatively cost-effective, while also satisfying a substantial proportion of the project's requirements. This quadrant is the "sweet spot" for identifying TCOs for products with high value.
- **Quadrant III** includes solutions that are relatively cheap, but do not satisfy many of the project's requirements. However, these solutions have potential if the organization wishes to spend more time and money on customization.
- **Quadrant IV** candidates should be eliminated from consideration. These potential solutions are comparatively expensive, and do not satisfy many of the project's requirements.

Figure 2 indicates which subsystems the observatory selected in bold.

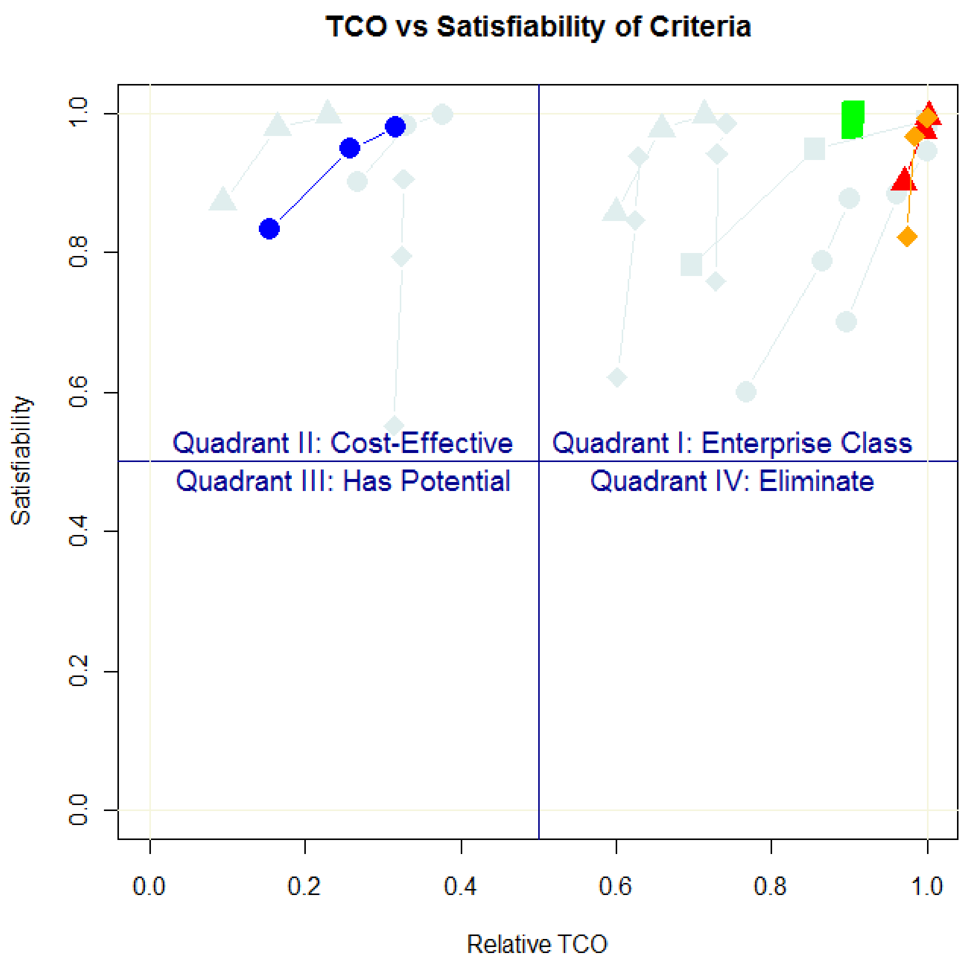

Figure 2. 3D TCO for the selected subsystems: asset management (triangles), messaging (circles), data processing (diamonds), and databases (squares).

3.3 Decision Criteria

Figures 1 and 2 were examined by the observatory in the context of their tentative product decisions. Upon examining the chart, several patterns were uncovered that informed the decision criteria for making trade-offs. These included: line segments that reach the top of the chart, indicating that all of the project's requirements can be satisfied; steep slopes, which indicate a large increase in satisfied requirements for a small investment over the lifecycle as reflected by the TCO; and line segments as far to the left as possible, in Quadrant II or ending in Quadrant II.

A fourth consideration is the length of the line segment. Short line segments indicate that there is not much difference between the work that is required to satisfy the highest priority requirements and all requirements (including the lowest priority ones). As a result, there is lower risk in selecting an investment alternative with a shorter line on the 3D TCO chart.

Analysis of Figure 2 shows clear reasons for selecting the messaging system (lowest TCO solution that meets the other value-based criteria) and the database (two comparable solutions, but one with much lower risk). However, this chart still indicates that the observatory has chosen an asset management system and data processing system that are expensive but do not provide greater compelling value. This picture changes significantly when the prospect of developing software systems from scratch is included in the analysis.

3.4 3D TCO with Custom Development Included

When COTS software is selected, there is always the possibility that some requirements of the project cannot be satisfied no matter how comprehensive the product. As a result, even a relative satisfiability score of 1.0 can mask requirements that go unfulfilled. When custom development is considered as part of the analysis, the relative satisfiability score of 1.0 does indicate that all of the requirements can be satisfied, because custom software does not have the limitations that COTS software may.

Figures 3 and 4 illustrate how the decision making process is adjusted when custom development is plotted as an option on the 3D TCO chart. Each of the line segments that represent custom development originates in the bottom left corner of the plot, because at the beginning of custom development, you have invested nothing and you have satisfied no requirements. The main difference between Figures 3 and 4, as compared to Figures 1 and 2, is that the maximum satisfiability has increased (because with custom development, it is always possible to fulfill all needs) and the maximum TCO may have also increased (if the cost of custom development exceeds COTS plus customization for one or more purchased products).

When the trade-offs between COTS products and custom developments are considered, a new pattern emerges. Although the messaging system and database still demonstrate clear value, it is also apparent that the data processing system (diamonds) and the asset management system (triangles) provide much greater value than developing those applications from scratch. Although these options appeared expensive in the first presentation of data, their relative merit is clear when compared to all options.

Figure 4 illustrates and enumerates the selected software subsystems.

4. CONSTRUCTING AND INTERPRETING A 3D TCO CHART

Following the example presented herein, the three dimensional value-based TCO chart considers the dimensions of satisfiability and priority of requirements in addition to the currency-based measures for TCO. Value is inferred by examining the position and orientation of the line segments on the 3D TCO chart. The chart is constructed by following these eight steps:

1. **Determine TCO Elements**: Examine the wide range of possibilities for cost elements that can be factored into your TCO calculations. Several of these are listed in Table 1. The costs should be selected so that, at the least, the major contributors to the acquisition and maintenance of the subsystem are included in the calculation. Minor contributors to the cost over time can be left out at the organization's discretion.

2. **Establish Time Horizon**: Choose a time horizon over which to sum the costs for each cost element you selected in Step 1. This should be an interval of time that has some meaning to your organization. For example, if your strategic planning cycle considers options 5 years into the future, then a 5 year time horizon may be appropriate for your TCO calculations.

3. **Prioritize Requirements**: This method assumes that you have partitioned the work to be done in terms of priority or some other factor (e.g. time, project phases, and development cycles), because value is inferred from how TCO increases over time as greater functionality is enabled in your project. If you have not partitioned the work, this must be done before the individual cost elements are calculated.

4. **Identify Proportion of Requirements Satisfied at Each Priority Level/Development Cycle/Phase**: The next step is to determine what proportion of the requirements are satisfied at each priority level (or in each development cycle or phase of your project, if you are using these instead of priority).

5. **Calculate TCO for Each Priority Level**: Using the cost elements identified in Step 1, add up all the costs for each candidate product or project *at each priority level*. For each candidate, this will produce an array of TCO values, with one TCO calculated at each priority level. It is these arrays that will form the basis for the line segments on the 3D TCO chart.

6. **Normalize Satisfiability Scores (Relative Satisfiability)**: Divide all of the satisfiability percentages by the highest possible satisfiability measure. This will yield a range of satisfiability scores between 0 and 1, which can then be plotted on the vertical axis paired with relative TCO measures.

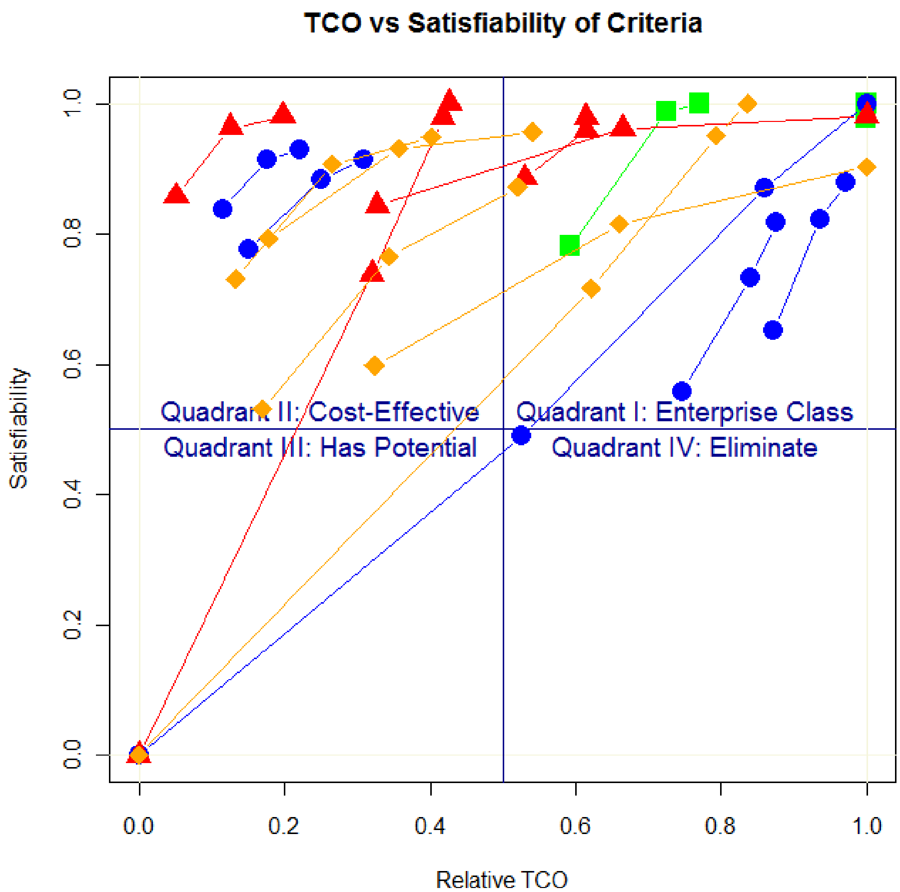

Figure 3. 3D TCO for the selected subsystems including a custom development option: asset management (triangles), messaging (circles), data processing (diamonds), and databases (squares).

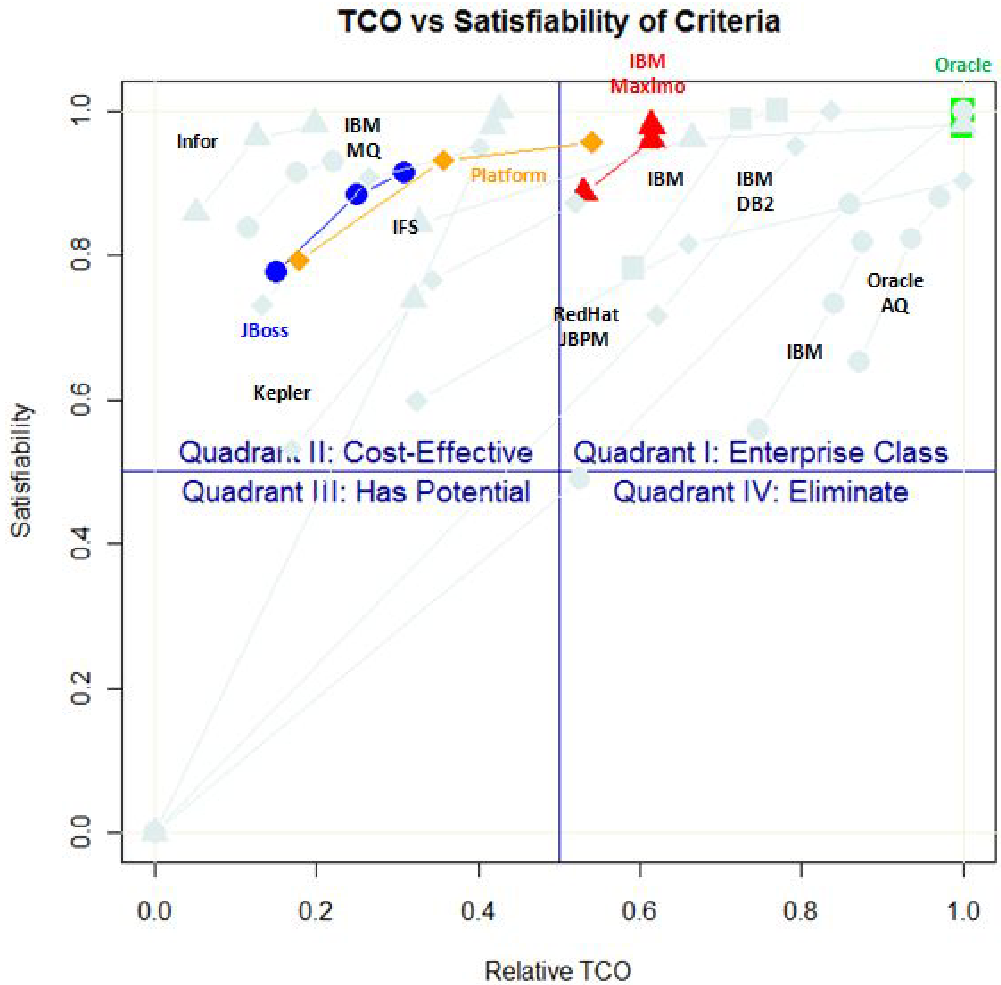

Figure 4. 3D TCO for the selected subsystems including a custom development option, with product candidates identified: asset management (triangles), messaging (circles), data processing (diamonds), and databases (squares).

7. **Normalize TCO Calculations (Relative TCO)**: Divide all of the TCO elements that will be represented on your chart by the largest possible TCO. This will yield a range of TCO measures between 0 and 1, which are plotted on the horizontal axis paired with normalized satisfiability scores.
8. **Construct 3D TCO Chart**: For each candidate product or project that you would like to evaluate trade-offs, collect one TCO calculation for each priority level. Plot normalized satisfiability scores on the vertical axis against normalized TCO scores on the horizontal axis, and connect the dots from the point at the lowest priority level to the highest priority level. The completed chart will have several line segments, each corresponding to one of the options for which you are examining trade-offs.

At this point, you will have a data set that consists of relative satisfiability and relative TCO for each product or development project that is being considered, assessed at each priority level for satisfying the requirements. The resulting 3D TCO chart can be examined for patterns that suggest which investment option provides the most *value* in terms of ability to satisfy requirements as the capabilities of the system are continually developed over time.

To evaluate trade-offs, examine the line segments that represent similar products or development projects. The options that deliver the most value can be identified using the following heuristics:

- **Line segments that reach the top of the chart (100% Relative Satisfiability)**: For any solution that is selected, it is important to ensure that that product choice has the potential to satisfy all of the project's requirements. Normalizing the percentage of requirements that can be satisfied by the maximum possible satisfiability ensures that you are comparing each option to the most comprehensive possible option, not to an unachievable ideal.
- **Steep slopes**: When a line segment has a steep slope, this indicates that a small incremental investment over the lifecycle (reflected in the increase in TCO) results in a large increase in additional functionality and ability to satisfy requirements. Typically, the slope will decrease as more and more functionality is added.
- **Line segments as far to the left as possible**: The line segments that remain farthest to the left on the chart will represent the smallest comparative TCO. Line segments that are far left, do not reach the top of the chart, and have shallow slopes can be interpreted as the "lowest bidder" solutions that may not actually be sustainable options for the organization to implement and deploy.
- **Short lines**. This indicates that there is not much difference between what it takes to satisfy your initial requirements, and what it takes to satisfy all requirements. As a result, there is a smaller risk associated with selecting the investment alternative. This, however, is the weakest of the four heuristics and should only be relied upon if qualitative considerations (e.g. quality, vendor support) are strong.

Although the case presented in this paper considered each software subsystem separately, it would also be possible to *batch* the subsystems together into different combinations. This method has not been attempted yet in practice, but would potentially capture systems integration costs more accurately, and provide a more accurate visualization of the trade-offs that would be required as product options are combined.

5. CONCLUSIONS

This paper presents an approach for visualizing currency-based Total Cost of Ownership (TCO) such that organizations can make strategic investment decisions based on value in addition to lifecycle costs. The method was devised to help a real observatory make strategic software decisions prior to the construction of instruments, and convince its funding agency that the lowest cost solution was not necessarily the solution that provided the greatest value.

The unique utility of this value-based approach is that it in contrast to other value-based methods, it does not use scoring mechanisms which alter the inherent meaning of the individual TCO calculations for each cost element. Rather, the approach uses currency-based TCO but partitions the data in such a way that value can be inferred from the visualization of the data.

REFERENCES

[1] http://dictionary.reference.com/browse/value
[2] Harriman, N. F., [Principles of Scientific Purchasing], McGraw-Hill, New York (1928).
[3] Ellram, L. M., "Total cost of ownership: an analysis approach for purchasing," International Journal of Physical Distribution & Logistics Management, 25(8), 4-23 (1995).
[4] Ellram, L. M., and Siferd, S. P., "Total cost of ownership: a key concept in strategic cost management decisions," Journal of Business Logistics, 19(1), 55-84 (1998).
[5] David, J. S., Schuff, D., and St. Louis, R., "Managing your IT total cost of ownership," Communications of the ACM, 45(1), 101-106 (2002).
[6] Bhutta, K. S., and Huq, F., "Supplier selection problem: a comparison of the total cost of ownership and analytic hierarchy process approaches," Supply Chain Management: An International Journal, 7(3), 126-135 (2002).

[7] BEA Systems, Inc. "Evaluating Total Cost of Ownership for Application Platform Suites", BEA White Paper. From http://www.crimson-consulting.com/marketing/experience/knowledge/white_papers/bea_hp_ibm_tco.pdf. (2003).

[8] Castellani, S., Grasso, A., O'Neill, J., and Tolmie, P., "Total cost of ownership: issues around reducing cost of support in a manufacturing organization case," Proc. 7th Intl. Conf. on E-Commerce Technology Workshops (CECW'05) (2005).

[9] Jones, C., "Geriatric issues of aging software," Crosstalk, December (2007).

[10] MacCormack, A., "The true costs of software," Computerworld, May 29 (2003).