Enterprise Applications Portfolio Management Utilizing COTS

Masafumi Kotani
IBM, Tokyo, Japan

mkotani@jp.ibm.com

Abstract

The software assets that an enterprise applies toward the management of its business processes are bound to evolve in response to business model transformations dictated by changing management environments. Both a given set of software and its associated data form an asset portfolio that needs to be well managed to generate the greatest possible return over time. Trade-offs between the TCO (Total Cost of Ownership, consisting of development and subsequent maintenance costs) and implementation speed present significant challenges to management decision-making. Here, a portfolio configuration management framework designed from a decision-maker's point of view is proposed, and an experimental evaluation result is reported.

Keywords: Business, IT alignment, Application portfolio management, COTS

Introduction

Enterprise applications and their associated data offer a crystallization of business processes whose cost-effort metrics, such as Function Point (FP) method, have been well defined over the past 35 years of software engineering and development. In a typical manufacturing enterprise of medium size, the applications portfolio is sized to average 200,000 Function Points of software, which translates to 8000 man-months of software engineering effort. Desktop spreadsheet programs, by contrast, are mere 50FPs.

Software development costs for enterprise applications occupy a substantial portion of investment decisions. As the investment size of information systems has grown, C-level (CEO/CFO/CIO) strategic decision-making for complex build-new, enhance-existing, and retire-obsolete processing hardware-software operations is necessitated.

Today, new product introductions must take place each quarter, minor adjustments in sales strategies are frequent, and strategic alliances for completing smooth-lined supply chains are forged. Such a management environment demands a rapid reflection of changing business processes in the implementation of information systems. Sales divisions require speedy updates of system services, but investment priority decisions often yield distorted service offerings to end users. As a result of limited information system budgets, this time-race is, all too often, not won. Practitioners often resort to budgets "proportional to revenue amount," but little correlation can be expected between "revenue size" and "system investment amount". Once an industry benchmarking of information system expenditures is considered to reveal a high correlation with "net earnings", a homogeneity of industry members can be established, but stakeholders are more likely to notice a lack of unique efforts on a management team's behalf. Return-on-investment (ROI) analyses of discrete IT projects have been worked upon by virtually all IT consulting firms, but a decision- making framework to tackle the issue of IT projects portfolios is nonetheless necessitated since decision-makers lack a tool for investment optimization that integrates discrete cost and return estimates with timing decisions.

Toward this purpose, to grasp the total project cost, we should include, for example, end-user activity-based cost improvements resulting from IT improvements. Additional traditional estimates of total cost of ownership (TCO) consist simply of development and maintenance costs. We should also valorize development speed in cost evaluation, considering the man-month equivalency of the time savings which results. Applying these considerations, software of traditional development projects and the large-scale commercially off-the-shelf software COTS can be applied to portfolio configuration decisions.

An estimation of investment returns requires templates for process definition in order to keep it immune from fluctuations caused by individual characteristics. We often witness the contingent fee bonus method for rewarding business strategy success. This suggest that the evaluation of project returns should be flexible to accommodate Real Option methods over and above the standard Net Present Value/ Discounted Cash flow valuation methods. With the Real Option method, we are able to valuate each selection of a decision tree in a high uncertainty environment, and furnish documented rationales for selections made. Applying Real Option Evaluation, we can limit the evaluation horizon to a few years for practicality and period of present-value, discounting accordingly. In addition, as this option is exercised, the retirement of related portfolio elements can be considered and decided. Real Options valuation can be applied to delaying, phased implementation, enhance/scale-down, abort, resume, disposition, combining project, and combining options. We will focus on project staggering and disposition decisions with DCF adjustment. Tax rule changes like the amortization method are excluded as an externality to the framework.

Enterprise Application Portfolio Management Framework

The application of investment development is split into 4 phases: research, planning, implementation, and evaluation/maintenance. Evaluation indices, which will be used throughout the project phase lifecycle, are defined in the research phase. Evaluation indices consist of both business and systems aspect parameters. This framework consists of benefit evaluation and portfolio optimization steps.

For expected benefit evaluation, we will utilize expected utility evaluation rules formulated by VonNeumann and Morgenstern (1944). An expected utility of project j is expressed as $EU_j = \sum_i P_i U(X_i)$ where P_i is the probability of outcome X_i, and $U(X_i)$ is the utility value of outcome X_i. $U(X_i)$ expresses two types of curves – convex for risk averse and concave for risk seeking. The maximum of each project utility EU_j is the key to the portfolio decision and EU_j indicates priority. After this step, Linear Programming is employed to single out an optimum project configuration. Execution steps of the evaluation framework are as follows:

Step 1. Assign number (i) to the indicator indices, which have been established in the research phase.

Step 2. Determine the weight of each indicator index.

Step 3. Assign probability (P_i).

Step 4. Assign contribution factor to each of indicator index ($U(X_i)$)

Step 5. Determine the expected utility for each project $EU_j = \sum_i P_i U(X_i)$

Step 6. Define the overall objective function for the portfolio $EU = \sum_j \beta_j EU_j$ where

β_j are the decision variables in the LP model which can take only binary (0,1) value.

Step 7. Calculate the total budget for the evaluation period $\sum_j X_i$ for resource types A and B ($X_A + X_B$).

Step 8. Determine the required investment for each project $\sum_{A,B} \beta X_{A,B}$

Step 9. Formulate the constraints for the LP model
$\sum_i \beta_i X_A(i) \le \max X_A$ and $\sum_i \beta_i X_B(i) \le \max X_B$

Step 10. By solving the above LP model, obtain decision variables $\beta_j = 0,1$ for each project. Backplane solvers of Excel or Lotus123 spread-sheet programs can be used for solving the LP model.

Cost-Time Characteristics of COTS-based Project

As is often the case, and as observed in our case study, when existing portfolio components indicate rapid recent year growth in maintenance costs, we employ a cubic-root t approximation for the maintenance cost estimate. Traditional application development projects use the Function Point method to estimate these development costs. For COTS projects, of the total COTS code structure, only utilized FP, are selected to estimate the development and maintenance workload. The license charge for COTS, and its maintenance costs are superior to own-development projects by the factor of 5 (see Figure 1). While COTS requires stringent risk evaluation, we understand that the functionality difference among COTS packages are becoming less significant, in terms of evaluations of customization ease, basic Fit-Gap analysis, and number of production users and their testimonials. For COTS, functional comparison has lost meaningfulness. Rather, we should focus on how to maximize utilization of delivered functionality and valuate the option value of rapid development capability.

Evaluation of Business Values and Risk

Industry-wide issues and problems, in addition to the unique situations of individual enterprise, dictates implementation target dates and functionalities to be realized. Reduction in processing cost of order acceptance and speed of fulfillment is mandatory for competitive reasons. Since business drivers are often clearly defined, evaluation indices were easily valued together with market size forecast and share gain estimates. Business risk is valued in terms of competitive entry timing that yields share gain-loss estimates.

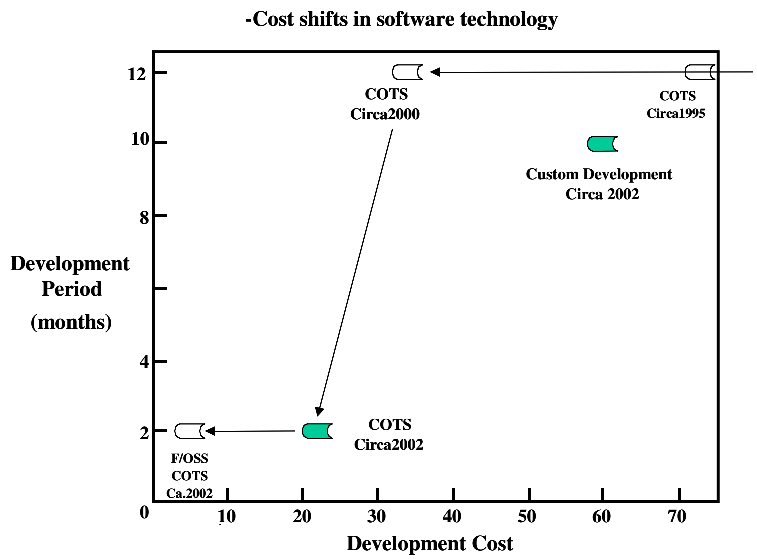

Figure 1. Comparison of COTS and custom development costs

Conclusions

Though limited to application portfolio management, a useable and repeatable decision-making framework has been suggested. Evaluation indices originated at the research phase have been used for value/risk scoring. With this framework, implemented by the Real Option method, decision makers are relieved from a difficultly managed lump sum, now-or-never budget decision and can pursue phased decision-making instead. Actual usage required 4 months to complete minor adjustments, but repetitive iterations will occur within a week of parameter evaluation.

Information technology advancement offers new options for information processing, such as Internet-based data center operations. Future work could include framework expansion and calibration to such emerging options. The framework also needs to be evaluated against small-medium scale industry situations typical in East-Asian regions where decision-making space is narrower because of financial and human-resource constraints.

Acknowledgement

This research was conducted as part of the consultation service that IBM / Encyclosoft offered to actual customer engagement situation. The author would like to thank to financial support by IBM.

References

Von Neumann, J. & Morgenstern, O. (1944). *Theory of games and economic behaviour*. Princeton University Press.

For Further Reading

Doherty, N. (2000). *Integrated risk management: Techniques and strategies for reducing risk*. McGraw-Hill.

Eick, S.G. & Mockus, A. (2000). Does code decay? Assessing the evidence from change management data. *IEEE Transactions on Software Engineering. 27* (1), 1-12.

ERP5. (2002). The software project list of existing products running on Linux. http://www.erp5.org/linux.html

Hissam, S.A., Seacord, R.C., & Lewis, G.A. (2002). Tutorial on building systems from commercial components. *Transactions of ICSE'02*, May 19-25, 2002, ACM, 679-680.

Kotani, M. (2002). Enterprise applications software trends and directions. *Open Software Programmers Group*, CIO Workshop Presentation, September 2002. OSPG.

METI. (2001). Government whitepaper: Challenges of economy policy for the 21st century. Ministry of Economy, Trade and Industry, Japan 53-58.

Sakaguchi, Y. (2002). After 21 years Mitsui Life renovated LOB system by developing 1,730,000 LOC. Nikkei Computer, November 2002, 68-72.

Trigeorgis, L. (1995). *Real options in capital investment: Models, strategies, and applications.* CT: Praeger Publishers.

Trigeorgis, L. (1996). *Real options: Managerial flexibility and strategy in resource allocation*. Cambridge, MA: MIT Press.

Biography

Masafumi Kotani is Marketing Executive of Industrial Sector of IBM Asia Pacific Service Corporation. His field of specialization is integrated risk management in business process reengineering.