

Can Projects Incorporate Too Little Risk? Part II

Description

OVERVIEW

In the previous post, I discussed an approach to valuing projects using Discounted Cash Flow (DCF) analysis. In this post, I will use the DCF technique combined with a fixed scenario to demonstrate a difference in Net Present Value (NPV) resulting from implementing a series of projects on an existing platform vs. replacing the aging platform and fulfilling the same requirements with up-to-date tools. The horizon of the analysis is 20 years.

The objective here is to illustrate how the “riskier” option of using a more current technology with which the organization may have less experience can pay off in the long run.

THE SCENARIO

Our company operates an application supporting a line of its business, written years ago in COBOL. We currently realize Net Income from the line of business of \$10 Million per year. A new product opportunity, which will increase our Net Income to \$12 Million, has become available and we must decide whether to enhance the COBOL application to support new required functionality or rebuild the entire application.

In addition, two other events are part of the scenario:

- In year six, new regulations will impose reporting requirements on our company that will require enhancing the application. There will be no net benefit for this but the company is obligated by law to meet these requirements.
- In year ten, we will be presented with a merger opportunity that can increase our annual Net Income to \$14.5 Million. It will require fairly substantial changes to the application and an increase in operating costs to support the merger.

GIVENS, ASSUMPTIONS AND OBSERVATIONS

The table, below, shows the capital (implementation costs) and operating expenses over the life of our analysis:

Year	Event	COBOL Capital Exp	COBOL Operating Exp	Rebuilt Capital Exp	Rebuilt Operating Exp	Net Income
0			2.50			10.00
1	New product opportunity	1.50	2.50	7.00	1.85	10.00
2-5			3.25		2.00	12.00

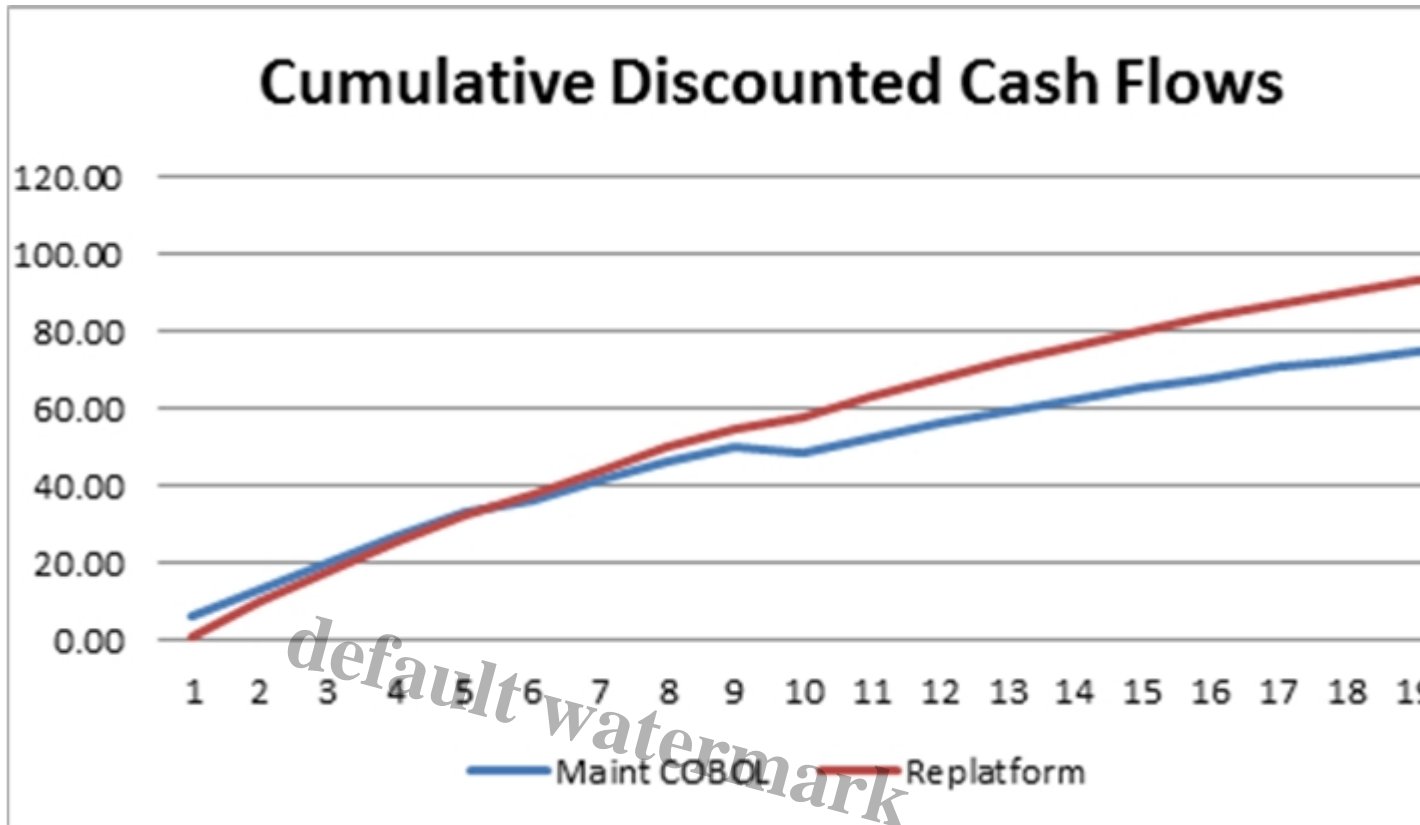
6	New reporting requirements	2.50	3.50	0.85	2.00	12.00
7-10			3.50		2.00	12.00
10	Merger opportunity	7.50	3.50	2.50	2.00	12.00
11-20			5.00		3.00	14.50

The data are designed with the following assumptions:

- It is cheaper to modify the COBOL application in year one to support the initial business opportunity than to rebuild it. It is cheaper to operate the rebuilt application, however.
- It is much cheaper to modify the rebuilt application to meet the new regulatory requirements in year 6 than to modify the COBOL application. I have assumed that operating costs for both platforms will remain level after this change.
- Similarly, it is much cheaper to support the merger in year 10 with the rebuilt application than the COBOL application. The difference in operating expenses widens between the two options at this point and remains substantial over the horizon of the analysis.
- The net business benefit of the line of business will be the same, regardless of which platform is used to support it.
- The riskiness of rebuilding the application is implicitly represented in the capital and operating costs of the new technology option. In the PMI approach to project estimation, reserves are created for implementation estimates for riskier work packages and they are added to the baseline cost estimates to provide the project a cushion for time or cost overruns.

MODEL RESULTS

The graph, below, shows the results of the Cumulative DCF over the 20-year life of the analysis.



The NPV of the investment in maintaining the COBOL-based application is \$76.61 Million and that of the rebuilt application is \$95.66 Million. Note that the COBOL option has higher returns initially but then becomes inferior to the replacement option. This results from the required investment in the replacement development project, which is subsequently repaid by the cheaper operating expenses and lower costs to make subsequent enhancements. In essence, the rebuilt platform has a lower Total Cost of Ownership (TCO) over its life.

DISCUSSION AND ANALYSIS

In the previous article, I identified three goals that increase the value of an application system:

- minimizing the number and scope of changes over its life,
- increasing its usable life and
- accelerating the onset of benefits.

In order to maintain the simplicity of the example, I accounted for easier and cheaper modifications for the rebuilt application but held the time to implement and achieve benefits constant for all three events in the scenario. Had I varied the implementation timeframes, I would have pushed back completion of the rebuilt application vs. the COBOL enhancements for the new product opportunity but accelerated the completion of the regulatory requirements and merger support events.

Similarly, I held the horizon of the scenario, 20 years, constant for both options. While it's hard to predict a usable life of 20 years for any application built today, there are still COBOL applications in use after 30 years, or more! Incorporating the decommissioning and replacement of the COBOL application at some point in the scenario is a realistic complication that I chose not to include for

simplicity's sake.

It's more than likely that the overall conclusion of the analysis would not change in response to either of these but the magnitude of the difference in NPVs might have.

Needless to say, there are many aspects of this example that are simplistic when compared with real life. Nonetheless, it is consistent with a number of precepts of Strategy and Enterprise Architecture:

- In the previous article, I pointed out that much of the value of an application system accrues later in its life, during a period of maturity in which the frequency of modifications decreases. This exactly parallels the lifecycle of a successful line of business, in which the organization competes vigorously in its early stages, outlasts competitors to gain a significant market share and then generates high returns, coexisting with only a few other large competitors.
- The use of newer technology should be gaited by the dynamism of the markets in which an organization competes. The greater the pace of change in the marketplace, the more agility and flexibility are required to maintain a company's position in it.
- Cost and benefit optimization across the enterprise is an important goal, which may supersede the results achievable for any individual project. While one set of tools and techniques may be best for a given case, it may not be optimal for the organization when compared with the option to reuse existing components, share an application system with another business unit or increase volume purchasing benefits. Not only may these options produce cost benefits, they may reduce implementation risks, as well.

CONCLUSION

In these two articles, I examined risks for application system projects in a larger context, using financial measures as a yardstick and factoring in the strategic impact of various options over the life of the assets created. What I have shown (I hope) is that choosing the least-risky approach for a project may not always be the best option and, therefore, it is possible that a development project may indeed have too little risk.

Date Created

2013/08/16

Author

howardmwiener