Universal Economic Analysis for IT Transformation

Chang-shing Perng, IBM Research, perng@us.ibm.com
Rong Chang, IBM Research, rong@us.ibm.com
Tao Tao, IBM Research, ttao@us.ibm.com
Edward So, IBM Research, edwardso@us.ibm.com
Mihwa Choi, IBM Global Services, mihwa@us.ibm.com
Hidayatullah Shaikh, IBM Research, hshaikh@us.ibm.com

***Abstract*—For any IT transformation with significant investment, economic analysis is essential for justifying the risks and expenditures. Much cost modeling has been done for various types of IT transformations, each looks into the unique aspects of the concerned domains. This paper takes a different direction and proposes a general methodology named UROI for quickly constructing ROI/TCO models to be used in the field.**

I. Introduction

A. Motivation

Economic analysis is essential for all IT transformations. All IT investments are made entirely or partially for cost saving or revenue increase. Return on Investment (ROI) and Total Cost of Ownership (TCO) analyses provide guidance on what and when to invest.

While the basic concepts and mathematical techniques of economic analysisare well studied[1], it is still not clear for IT practioners on how to quickly construct a model, implement it as a tool, and deliver it to end users.

B. Use of ROI Analysis

ROI analysis are conducted in different phases.

1) In the *pre-sale stage*, the stake owners would like to understand how a particular type of IT transformation can financially benefit their organization. Without pulling out a complete inventory list and a detailed business process mapping, one can conduct ROI analysis with a rough estimate of the infrastructure. For example, *what is the ROI for moving workload from* 500 *self-own and self-managed servers to a public cloud?* The goal of pre-sale ROI analysis is to let a prospect understand the investments and the sources of saving.
2) Once the prospect shows interest in taking such IT transformation, the *engagement stage* starts. In this stage, the stake owners can provide more detailed information of the IT infrastructure including the list of hardware, software, facilities, and employees involved. The type of questions to be answered is like: *what is the ROI for moving development and test workload used by* 50 *software engineers from* 250 *Windows server and* 250 *Linux servers managed by* 10 *system administrators?* The goal of engagement stage ROI analysis is to provide decision support for going forward with the IT transformation or not, and for choosing technology and services vendors.
3) Once the stake owners commit, an on-boarding stage ROI analysis can be conducted to find the optimal transformation process and schedule. The focus is to decide what workload should be migrated at what time.

C. Challenges

In practice, building economic analysis models is not a trivial task. The difficulties come from

- **Interdisciplinarity**: Building financial models for IT transformation is a task that requires knowledge on both IT operation and financial modeling. Few institutes offer such training. The norm in the industry is to have IT professionals to learn financial modeling and build such models instead of having financial experts to learn about IT operation.
- **Lack of Tool Support**: There are very few, if any, tools that provide guidance for building economic models. Most commonly, a model builder starts from a blank spreadsheet, then struggles to identify cost items, formulate future costs, and create a friendly user interface all at the same time. Once such an analysis tool is delivered to end users, it is very hard for the model builder to collect actual data in order to gain more insight for further improvement.
- **Model Complexity**: The underline logic of most economic models, if expressed pure mathematically, can be easily comprehended. However, when it is implemented as a spreadsheet application, the number of cells quickly goes out of control. To make the matter worse, cells are mostly referred by coordinates and the formulas are not visible unless the cursor is moved to the cell. These factors make understanding and debugging a models very difficult, and hence make it a very expensive cost item itself. An even more serious problem is that a complex model become a black-box to the decision makers and weaken the ability to convince them.
- **Adaptivity**: It is very common that an economic model is created by one group and used by another. Usually model creators can not foresee all possible situations. The end users should be able to change and fine-tune models. Most existing models are either too complex for the end users to change, or the way of delivery, e.g. a web or desktop application, does not allow end users to change the models. We have seen many costly built models abandoned because of some minor mismatches between the actual situation and the model. It is very important to provide End User Development[2] capability.

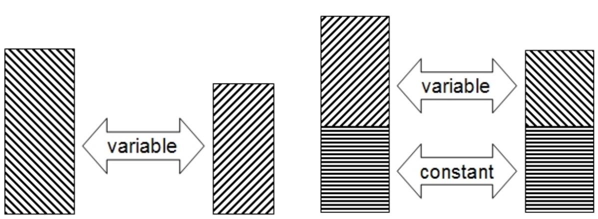
II. The UROI Methodology

A. Principles

In terms of scoping, there is a difference in the principles of Total Cost of Ownership Analysis and Return on Investment Analysis. TCO analysis, as the name implies, include all the related cost items in the model to produce an estimate of the total expenditure required. In contrast, to conduct an ROI analysis, the focus is on the additional investments outside existing planned expenditure and the anticipated additional benefits. Cost items that remain constant throughout the transformation are not of interest and should be excluded from analysis. To reconcile the differences in the two principles, we propose a trade-off between the exhaustive but time consuming TCO analysis and the simple but selectively focused ROI analysis.

Our approach is to define a set of core cost categories for which we calculate the total cost. In addition to the core cost categories, we define a set of benefit categories for which we only compare differences among the alternatives. This is illustrated in Fig. 1. For example, consider the case of migrating workload running on self-owned infrastructure to a cloud that provides virtual machines bundled with operating systems. In such cases, we consider hardware and OS software as core cost categories since it makes more sense to estimate the cost items that will be eliminated. As for the labor cost of business process, it is usually harder to define the boundary of personnel involved in a particular business process so we generally consider it as a marginal category so we can focus only on the increased productivity and labor reduction brought by the cloud instead of surveying all labor involved in the whole business process.

In terms of model building, we propose a top-down approach. Starting with understanding the stage of IT transformation discussion, choose a data granularity for the model, then work the way down to formulate the cost structure and project future cost. The UROI methodology guides model builders to build economic models that are adequate for decision support.



(a) Core cost type (b) Marginal cost type

Fig. 1. Mixture of ROI and TCO

B. Model Construction Steps

The 7 steps of UROI methodology are shown in Fig. 2.

1) Positioning: Economic analysis models are used for different purposes, consumed by different audiences, in different decision stages. So the first step of model building is to position the model by understanding the following issues.

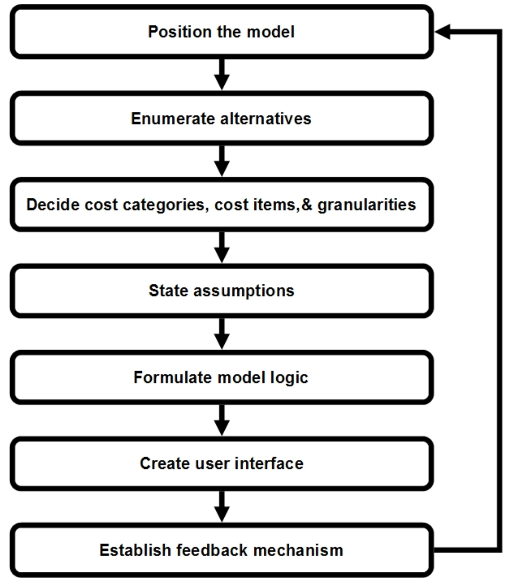


Fig. 2. Steps of the UROI Methodology

- **Purposes**: What are the goals to be achieved? Is it to support major investment decision, estimate total cost for budgeting purpose, or to compare solutions and vendors?
- **Target users**: Is the analysis to be conducted by professionals who have been trained to use the model, or by customers who have no prior experience in economic analysis?
- **Target audiences**: Is the analysis to be consumed by executives, or by project managers
- Time for analysis: Is the analysis to be done in pre-sale, engagement, or on-boarding stage?
- Available data: What granularity of the data would the target users have?

2) Enumerate Alternatives: The goal of UROI is to compare a number of alternatives to see which yields the most benefits comparing to the existing IT operation model. The existing operation model, following the convention of statistics, is called the **null alternative**. Note that a null alternative does not imply the cash flow or capital spending is constant.

An alternative is an operation model that is qualitatively or quantitatively different from the null alternative. Examples of alternatives are like replacing a supply chain management software with a software-as-a-service, outsource a customer help desk, or host a web service in a public cloud.

3) Decide Cost Categories, Cost Items, and Granularity: Cost categories, each contains some cost items, are used to structure the overall cost, and to provide a way for comparing alternatives. For example, one can define a ”Hardware” cost category which contains servers, storage, and networking equipments, then compare the hardware costs of each alternatives.

A cost item is an asset or a service that require capital for acquisition or maintenance. Each cost item can be characterized by its charging schedules. There are basically four types of charging schedules. i) One-time charge, e.g. purchase of real estate. ii) Periodic charge, e.g. Annual service charges, rent. iii) Replenish charge, e.g. equipments that have limited life iv) Usage charge, e.g. energy and water. Each cost item is charged by one or more of the charging schedules.

Common cost item types include the followings.

Tangible Asset: Tangible assets in IT are hardware assets that directly serve computing need. Common tangible assets include servers, desktops, storage, and networking equipments. Assets are usually charged for purchase cost.

Software License: Software licenses are charged in many ways. System software is usually charged by the hardware configuration it runs on. It can be based on number of computers, number of CPU sockets, or Processor Value Units (PVU)[3]. End user software, e.g. development tools, is usually charged by the number of named or floating user seats. Some software is charged only once at purchase time. Most enterprise software charge also include support services costs which are billed periodically. In cloud, software may be charged hourly.

Labor: Labor cost is increasingly important to IT transformation. In the many situations involving cloud computing, labor cost becomes the determining factor as one of the promise of cloud computing is to greatly reduce labor and time required for acquiring and releasing computing resource. Usually, there are many types of labor involved in IT operation, including those administrate the IT system, those manage business processes, and those develop and test applications. Depending on the granularity to be used, labor can be roughly estimated using one average labor rate as one cost item, or be separated to a number of cost items for more accuracy.

Services: The connotation of *services* has been broadened greatly in the recent years. Traditionally, subscribing to service mean to be served by human professionals. In the age of web services and cloud computing, services can be delivered with no direct human involvement in the process. For example, software-as-a-service, platform-as-a-service, and infrastructure-as-a-service are all considered services. Services can be charged in all possible ways. The majority of services are charged periodically, services like software installation services are one-time charge. There are also services that are charged by usage.

Utilities: For data centers, utilities like electricity, fuel, and water can be significant expenses. In many cases, energy cost dominate the decision factor for data center location selection.

Others: Depending on the type of IT transformation of interest, there might be many other cost items to be considered like power equipments, site security. Sometimes, tax and aide are also included.

Decide Granularity: Based on the stages of the sales cycle and the availability of data, we can choose to use different granularities for modeling assets and resources.

1) **Total** is the roughest granularity. The model only uses the total number of the resources. For example, use total number of servers as the input regardless of the sizes of the servers and the operating systems, use total number.
2) **Categorized Total** improves accuracy by allowing assets to be classified to multiple asset classes. For example, servers can be divided to large Unix servers,small Windows servers, etc., then use the total of each asset class as input.
3) **Detailed** uses the actual list of cost items as input which contains the detailed information of each item. For example, use server inventory list with every server's capacity, energy consumption, operating system, and age for hardware input, use actual skill and pay rate for each employee.

C. State assumptions

Economic models are built to project the future. Many assumptions have to be made in order to make concrete formulation possible. No model can include all possible variations of the IT transformation; a trade-off has to be made to keep the model comprehensible enough for users. This has to be done by clearly stating all the assumptions made for the model. For example, a model for projecting workload and computing power demand growth usually need an assumption that such growth is at a constant rate.

Assumptions also serve as a way to screen the applicability of the model, only the situations conform to the assumptions can be analyzed by the model. Also, having a clearly stated list of assumptions enable model builders to validate whether the formulation of the model is compatible to the assumptions.

D. Formulate model logic

The core step of the methodology is to actually formulate the mathematical relationship between input data and results. The steps include:

- **Collect available data**: A model usually requires some constant values built into the model. Examples include exchange rate, average wage, utility cost, service price, etc.
- **Identify input and output variables**: Based on the cost categories and cost items already defined, a model builder should first enumerate the output variables like the total cost, the net present value of an alternative, the subtotal of a cost category. Then create formulae and see what are required inputs.
- **Model transition**: IT transformation is usually a process that can span from a few months to a few years. Extra attention has to be paid on how the cost would incur from one operation model to an alternative.

E. Create User Interface

Once the model logic is created, it requires an user interface to accept input data and present the results. A user interface has to be intuitive and make the meaning of all questions and results very clear to users. A common practice is to have a section for input data, a section to present the end results, and section to show how the results are calculated from the input data.

F. Establish feedback mechanism

Once a model is deployed, it is essential to gather feedback for the following purposes:

- **To tune model**: For any model at modest complexity, there is always a chance for error and inaccuracy. End users can usually spot bugs quickly.
- **To collect industry averages**: Some numeric values might not be known at the model building time. As actually data are collected, one can establish a baseline for better understanding the average cases. In practice, it is common to collect data to establish baseline for specific industries (financial, telecommunication, parcel services, etc) and use the baseline to reduce the number of mandatory input required.
- **To identify usability issues**: Users can usually provide valuable feedback on how to make the user interface more intuitive and understandable.

Feedback mechanism can be implemented by business rules or technology. Cooperates can demand model users to send back data and comments. This can apply to tools implemented in any form. User feedback can also be a built-in function of the tool that sends back data automatically to a central location.

RELATED WORK

The white paper from Berkeley[4] explained the quantitative and qualitative benefits of generic Cloud Computing excluding the cases for private clouds. [5] discussed the cost of Cloud with special focus on network, power, and servers. [6] provided a detailed account for a cloud migration with discussion of the benefits and risks of such migration. [7] use an astronomy application *Montage* as an example to analyze the benefits of scientific computation using Amazon EC2. There is very little, if any, published work that discuss methodologies for building economic analysis models for cloud computing or other IT transformations.

The financial theory for building economic analysis model has been fully developed. The key concepts like cash flows, interest, rate of return, and net present value are discussed in Newnan et al[1] and the like. While the mathematical foundation has been well-established, it remains challenging to construct a usable model because the demand of domain specific knowledge and the lack of tool support. Much work has been done for building specific types of IT economic analysis models. For example, Reifer [8] present a methodology for building business case for software.

This motivation of this paper is largely from meeting the need for building cloud computing ROI models. It has been a very prominent topic in academia, IT-related professional magazines and blogsphere. Armbrust et al.[4] outlined why cloud computing is financially beneficial considering the scale it can achieve. Deeman et al.[7] reported the experience of using cloud for scientific computing. Greenberg et al. [5] focused on the cost of networking in data center. Tak et al. [9] concondicted an analysis on comparing in-house and cloud on running workload similar to TPC-W. They covered cost categores including servers, storage, OS, and bandwidth, and showed there is a equilibrium point where the sum of provisioning cost and bandwidth cost is the minimum. Pisello [10] has a series of articles discussing the costs and benefits of various IT investments, from server virtualization to adoption of Service-oriented Architecture. Hamilton's blog [11] contains rich information related to data center cost and efficiency optimization.

III. CONCLUSION

The drive of most of the IT transformations to seek out efficiency, flexibility, and resilience can usually be reduced to financial benefits. Hence engineering economic analysis has always been critical in making IT investment decisions. However, building a correct and user friendly economic analysis model has not been an easy task even for IT professionals. The lack of methodology and tool support are the two major obstacles to be conquered. This paper aims to provide a general solution for building economic analysis models for IT transformation.

We introduce a methodology and tool support for building general economic analysis. The methodology starts with positioning the model. Then enumerates all the alternatives to be considered for analysis. The next steps is to decide the cost categories for comparing subtotal cost among alternatives, the cost items and their granularities for formulation. Next is to clearly state all the assumptions made for projecting future cash flows. Then one can proceed to formulate the logic model that captures and projects cash flows of all alternatives. With a sound logic model, the next task is to create a user-friendly interface that accepts input and deliver results. The last step is to establish a feedback mechanism that gathers user feedback for later assumption adjustment and model tuning.

REFERENCES

[1] D. G. Newnan, J. P. Lavelle, and T. G. Eschenbach, *Engineering Economic analysis*. Oxford Uniersity Press, 2009.

[2] H. Lieberman, F. Patern, and V. Wulf, Eds., *End User Development*. Springer, 2006.

[3] "Processor value unit," http://en.wikipedia.org/wiki/Processor_Value_Unit.

[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[5] A. G. Greenberg, J. R. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: research problems in data center networks," *Computer Communication Review*, vol. 39, no. 1, pp. 68–73, 2009.

[6] A. Khajeh-Hosseini, D. Greenwood, and I. Sommerville, "Cloud migration: A case study of migrating an enterprise it system to iaas," *CoRR*, vol. abs/1002.3492, 2010.

[7] E. Deelman, G. Singh, M. Livny, G. B. Berriman, and J. Good, "The cost of doing science on the cloud: the montage example," in *SC*, 2008, p. 50.

[8] D. J. Reifer, *Making the Software Business Case*. Addison Wesley, 2002.

[9] B. C. Tak, bhuvan Urgaonkar, and A. Sivasubramainam, "To move or not to move: The economics of cloud computing," in *The 3rd USENIX Workshop on Hot Topics in Cloud Computing*, 2011.

[10] T. Pisello, "The roi guy," http://blog.alinean.com, 2011.

[11] J. Hamilton, "Prospectives, james hamilton," http://perspectives.mvdirona.com/.