Decision Modelling and Optimization for Enterprise Migration to Clouds

By

Alexander C.B. Ng

A thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements for the
MSc. degree in
Electronic Business Technologies

University of Ottawa

Abstract

Many enterprises are currently exploring the possibility of migrating some or all of their IT functionalities to public clouds with the objective of reducing their overall IT service costs or to open new business frontiers. Unfortunately, making such a decision is not a straightforward task; it requires a vigorous evaluation of the various benefits, risks and costs associated with the migration of their diverse business processes that comprise of their current IT services. Yet, this problem has received very little attention in the literature, mainly due to its interdisciplinary nature.

This thesis aims at filling this gap by aiding the enterprises during the phase of making their cloud migration decision. The contributions of this work are twofold. First, a novel cloud-migration framework is introduced to guide the enterprises through a sequence of well-defined recommended analysis steps. These steps culminate with the formulation of the migration decision problem as a mathematical optimization one. The second contribution is a decision engine that efficiently solves this optimization problem.

More precisely, the proposed framework gradually guides the enterprise to first identify the various business processes that are related to their IT services and then to determine the relationship and the communication needed among those processes. The identified inter-process communication represents an indicator of how tightly coupled these business processes are to each other. When outsourcing business processes, tightly coupled processes add a high communication cost and may introduce service latency if they are not co-located. As such, inter-process communication becomes an important input parameter that affects the migration decision.

Enterprises can then determine to partially or completely migrate IT services to clouds. Furthermore, multiple vendors can be used for different services. However, when different vendors are involved, the communication cost between different processes increases. The objective is to maximize profit for an organization which includes lowering IT expenses in the long term without compromising data integrity or security. An optimization formula is finally constructed to help the enterprise determine which services to migrate given input parameters of the cost of doing business in-house, cost of outsourcing, and communication costs.

Finally, a case study is utilized to demonstrate the performance of the proposed work by analyzing the process of migrating the services to clouds for an IPTV service provider. More specifically, the case study focuses on the content delivery network (CDN) within the IPTV provider's infrastructure which is responsible for delivering contents to viewers. The CDN network can use the proposed profit-optimization formula to determine whether to utilize a cloud service or to use its internal resource to deliver the content. A performance evaluation from a simulation is presented to demonstrate the proposed profit-optimization formula can return a set of optimal mix of both internal and external services to maximize profits.

Acknowledgements

I would like to thank my supervisor Dr. Nancy Samaan for her guidance and encouragement. And I also want to thank Dr. Dominque Ferrand for his guidance while he was teaching Research Methods in Electronic Business Technologies course.

Table of Contents

List of Tables

List of Figures

Acronyms and Definitions

Acronyms	Definition
CapEx	Capital Expenditure
CCM	Cloud Cube Model
CDN	Content Delivery Network
CRM	Customer Relationship Management
ETL	Extract, Transform, and Load
IaaS	Infrastructure as a Service
IPTV	Internet Protocol Television
IS	Information Service
IT	Information Technology
ITIL	IT Infrastructure Library
PaaS	Platform as a Service
SaaS	Software as a Service
SLA	Service Level Agreement
VD	Virtual Desktop
VoD	Video on Demand
VM	Virtual Machine
VoIP	Voice over Internet Protocol

Chapter 1 Introduction

Outsourcing information technology (IT) services to contractors has been a common practice in various industries for many years. Nonetheless, the term outsourcing has evolved a lot over the past few decades; from a general perspective, outsourcing refers to services, that were traditionally performed inside an enterprise, being performed by external service providers. At a systems level, the broad definition of outsourcing is the practice of turning over part or all of an organization's information system (IS) functionalities to external service provider(s) [1].

From the above broad definition, the use of public cloud computing is a form of outsourcing where services can be outsourced to cloud vendors at the software, platform, or the infrastructure levels. While there are numerous articles that focus on developing efficient outsourcing strategies, the current literature lacks key solutions at an enterprise level that aim at optimizing the migration decision of existing IT services to the cloud.

There is a need to investigate the various migration strategies to identify which IT services to migrate to clouds at an enterprise level by creating a framework to map business objectives to business processes, and then from business processes to IT services. Once the mapping is defined, the enterprise requires decision analysis with the appropriate tools to determine which business processes and/or IT services to migrate to clouds to maximize the net enterprise profit.

1.1 Business Value of Migrating to Cloud Computing

There is a growing pressure from executives to reduce IT expenditures. These expenditures typically consist of the cost of hardware and network resources, hardware upgrading costs, software development/configuration, and salaries of IT staff to maintain these resources. Through cloud computing, hardware and software resources costs are reduced. This cost reduction is mainly achieved by the fact that all cloud users share the cloud resources and only pay for the amount of resources that they are using.

The sharing concept is realized through resource virtualization [2]. Using virtualization, a single physical resource is virtualized to create multiple different virtual resources that can be shared by multiple users or processes. For example, at the Infrastructure as a Service (IaaS) level, the concept of virtualization is used to host multiple tenants on a single server [2]. In this sense, virtualization allows companies to put multiple virtual machines into a single server rather than having to spread and maintain multiple computers, therefore, virtualized servers will have a long-term cost savings as a smaller number of resources are shared rather than maintaining a larger number of standalone servers. Furthermore, as this server is hosted on a cloud, the enterprise will only have to pay for the price of its shared use rather than for its purchase and maintenance costs.

This motivation has positioned cloud computing migration issues as the main focus of personnel at the executive level in enterprises where they aim at developing efficient migration strategies that can leverage cloud computing resources to reduce IT expenditures.

According to a survey conducted by CIO magazine [3], 38% of business executives make cloud computing a high or critical priority and 18% have budget allocation for cloud computing services. This trend to migrate services to the cloud is gaining momentum; one indicator of this trend is the increase in revenue of cloud vendors [4].

According to IDC Cloud, worldwide revenue from public IT cloud services exceeded $21.5 billion in 2010 and will reach $72.9 billion in 2015 [5].

However, migrating from the traditional silo/departmental design, where each department owns their software, to a cloud computing model is not a straight forward task. A cloud migration decision framework and an optimization strategy can help enterprises decide which services to migrate to clouds and which services to maintain in-house to maximize their profit. The initiative to move services to the cloud is not solely an IT decision; it requires collaboration from both the IT and business departments.

1.2 Thesis Statement and Primary Research Objectives

In many enterprises, the IT department is responsible for migrating applications into the cloud; however, various business departments own business processes. The decision to migrate to cloud is both a business and an IT initiative; it is important to define the objectives for each business process whereas the IT department can map different IT services to each business process [6]. A business process uses one or more IT services to accomplish its objective; and a software application is a composition of multiple business processes. For example, the business objective of providing video over IP (IPTV) to end users is to provide video content in a timely manner. One example of an IPTV service is video-on-demand (VoD) service offered by cable companies or online video providers. VoD business processes (from a provider perspective) consist of content management and content delivery [7]. The purpose of the content management process is to fulfill video requests from viewers by storing contents from various sources. The content delivery process is to identify which content servers store the requested video to be delivered within the service level agreement (SLA) timeframe and to deliver the content from the content server to the end user's set-top box or computer. The content management business process consists of two IT services: i) disk storage and ii) an inventory database to store which servers contain this video content. When there is a high demand of providing video streaming, it creates a load on the content-delivery server. One cost-efficient method of handling sporadic demand to offset server load is to use cloud computing to bring up a new virtual content-delivery server (based on pay-per-use pricing model) when load is above a certain threshold.

There is a lack of research contributions in the area of cloud migration decision modelling for enterprises. The objective of this thesis is to propose a framework that can be used to guide enterprises to decide which services to migrate to clouds using a profit-based optimization model. The framework steps include the possibility of redesigning existing processes to leverage the use of SaaS. If SaaS is used, IT services that are mapped to the business processes will also be migrated to cloud applications.

This thesis also establishes the necessity to use business processes as the migration units themselves rather than limiting the migration decision directly to IT applications. Hence, the transition to the cloud using the proposed framework also gives the enterprises an opportunity to rethink, at the requirement level, what business should deliver so that business and IT departments can collaborate on process optimization/redesign (the "how") that are tailored to the business requirements with cloud-based solution in mind, and to shed unnecessary processes that do not generate value.

1.3 Benefits of Cloud Computing

The benefits of cloud computing are:

- **Lower costs** - Rather than purchasing or leasing equipment that may not be fully utilized at all times, via Cloud Computing, customers can reduce the amount of physical space required and costs associated with developing, testing and maintaining new or existing systems. Lowering costs is the main driver for moving to the cloud [8].

- **Faster deployment speed** - Cloud offers lower upfront setup costs and faster speed of deployment. Cloud enables businesses to roll out solutions much quicker than with on-premise because with SaaS there is no need to procure hardware. With faster deployment speed, cloud applications drive faster Time-To-Value [9].

- **Reduced support needs** - Through SaaS, cloud clients often can reduce or eliminate IT support. SaaS providers typically include help desk support. Technical support needs are lower since SaaS providers do the patching and bug fixing of their software [9].

- **Simpler, more frequent upgrades** - Cloud applications offer seamless, automatic upgrades; this means users always have access to the latest features and functionality [9].

- **Better utilization** - Pay-as-you-go applications typically yield better adoption because firms pay for what they need eliminating the "shelfware" problem; and cloud vendors often deliver proactive health check reports that provide statistics about usage, making it easier for enterprises to identify usage patterns [9].

- **Opportunity cost** - According to HP, enterprises can spend upwards of 70 percent of annual IT budgets on operations and maintenance which means enterprises are spending 30 percent or less on innovations that are critical to grow their business [10].

- **Standardization of processes** - Cloud computing encourages IT organizations and providers to increase standardization of protocols and processes so that components of the cloud computing model can interoperate properly and efficiently [9]. The benefit of a standardized process is that it takes less time for system integration.

- **Scalability and better alignment of technology resources** - An organization can scale up and down its capacity from one server to hundreds of servers without capital expenditures. This ability enables an organization to obtain large amounts of computing resources to perform temporary computing-intensive tasks whenever required without investing in excess computing capacity to meet infrequent high-demand periods. Cloud computing's scalability is beneficial to projects that require vast amounts of storage or processing capacity for limited times [11].

- **Environmental benefits (Green IT)** - Cloud computing benefits the environment by reducing the overall power consumption, carbon emissions, and physical land use [11].

- **Decrease effort in managing technology** - Cloud computing eases the burden on IT resources for organizations facing shortages of qualified IT professionals.

- **Collaboration** - Cloud computing provides an environment where consumers can develop software-based services that enhances collaboration and fosters greater information sharing [12].

- **Customization** - With cloud computing, specific processes can be easily altered to meet changing needs through configuration, with no need to redevelop from the back-end systems [12].

1.4 Challenges of Cloud Computing

This thesis acknowledge that there are security and privacy concerns in cloud computing. The cloud concerns include reliability of cloud vendor's network, stability of cloud vendor, data security when information is saved in the cloud, data integrity, and data confidentiality [13]. This thesis focuses on the decision modelling and optimization for cloud migration where cloud security concerns are assumed risks.

1.5 Motivation

The analysis phase of cloud migration decision is important and worth doing because cloud services are beyond just at the IaaS level; that is, cloud computing is more than just virtual machines and remote data storage. To fully take advantage of cloud services to lower IT expenditure, businesses should look at all three level of cloud services: Software as a Service (SaaS), Platform as a Service (PaaS), and at Infrastructure as a Service (IaaS). An optimization model can help businesses decide amongst a set of related services which ones to migrate to clouds. This thesis addresses questions that enterprises have while migrating services to the cloud.

There are still many issues relating to cloud migration such as which deployment model (public, private, or hybrid cloud) should an enterprise pick for its services? How to select cloud vendors? Which cloud service pricing strategy is right for my business? How to optimally transfer risks and design SLA contract with cloud vendors? How to design an optimal set of rules to implement the adoption of a private or public cloud by an organization? How to manage and implement consistent IT policy across the usage of multiple cloud providers? How to design an optimal set of rules to decide the types of applications that should be moved to a cloud? What are security standards and issues of cloud computing? How to develop a methodology to assess risk from adopting cloud computing [13]. This thesis addresses questions concerning selecting which cloud services can satisfy the business objectives with a series of steps outlined in a proposed framework. This proposed framework suggests enterprises to develop a cloud migration strategy, to define inter-dependencies of their business processes, and to calculate the per-unit costs based on sales forecast which will be used as inputs into an optimization function to determine which vendors are the most cost efficient. During literature study, there are articles relating to calculating a monetary value for risks associated with cloud migration.

1.6 Methodology Evaluation

This thesis is based on a quantitative research methodology through literatures relating to IT service models, outsourcing cost models, and cloud migration. Comparisons of different cost models and how costs are categorized were described. Based on review of current literature, this thesis proposes a framework and an optimization model for enterprises to analyze their existing business processes and to evaluate cloud alternatives to increase profits. This thesis uses an empirical validation research method by modeling the migration problem in a form of revenue and expense functions. A simulation of different cost functions is used to validate the optimization decision function can determine an optimal set of internal and cloud IT services that maximizes the net profit.

1.7 Contributions

Enterprises have existing IT applications serving business processes that are hosted in servers within their data centers which are part of their overall IT expenditures. To lower IT expenditures, cloud-based solutions are sought. The decision to determine which IT services to migrate to the cloud and what percentages is not a simple task as there are many criteria to consider. Migration decision requires a vigorous evaluation of benefits, risks, and costs associated with migrating these business processes and their IT services to the cloud. This migration decision problem has received very little attention in literature research due to its interdisciplinary nature.

This thesis aims at aiding the enterprises during the phase of making their cloud migration decision. The contributions of this thesis consist of a novel cloud migration decision framework, and a decision engine that efficiently solves this optimization problem. Firstly, this thesis proposes a novel framework for cloud migration decision to guide enterprises through a sequence of well-defined recommended analysis steps. The steps outlined in the proposed framework culminate with the formulation of the migration problem as a mathematical optimization one. Secondly, this thesis proposes a migration decision engine that efficiently solves this optimization problem.

The objective of the proposed framework is to gradually guide the enterprise to first identify the various business processes that are related to their IT services and then to determine the relationship and the communication needed among those processes. The analysis of the inter-process communication is important in identifying how tightly coupled these business processes are to each other. When outsourcing business processes, tightly coupled processes add a high communication cost and may introduce service latency if they are not co-located. As such, inter-process communication becomes an important input parameter that affects the migration decision.

Furthermore, multiple vendors can be used for different services. However, when different vendors are involved, communication costs between different processes increase. The objective is to maximize profit for an organization which includes lowering IT expenses in the long term without compromising data integrity or security. An optimization formula is constructed to help the enterprise determine which services to migrate given input parameters of the cost of doing business in-house, cost of outsourcing, and communication costs. The objective of the proposed migration-decision engine is to determine the optimal quantity of IT services to partially or completely migrate to clouds that will maximize profit of this optimization formula.

An IPTV case study is utilized to demonstrate that some services in the content delivery network (CDN) can be outsourced to the cloud. This case study is used to explain how to apply the steps outlined in the proposed framework to help IPTV service provider to decide which internal IT services and how much to migrate to cloud services. Finally, a performance evaluation of a simulation using Matlab is drawn to show that an optimization formula can help enterprises maximize their profits.

1.8 Thesis Organization

The remainder of the thesis is organized as follows:

- Chapter 2 provides a detailed overview of existing IT service models and the types of outsourcing costs associated with these services. Outsourcing approaches and their limitation are also discussed in this chapter;
- Chapter 3 provides an overview of the cloud computing paradigm, service models, deployment models, and pricing models;
- Chapter 4 provides a literature-based review of the differences and similarities between outsourcing and cloud migration, and demonstrates that cloud migration can adapt outsourcing decision-making techniques;
- Chapter 5 proposes a novel framework that can be used to guide the enterprises through a sequence of steps to formulate their migration decision problem as a corresponding mathematical optimization formula. The chapter then proposes an optimization engine to solve the developed formula;
- Chapter 6 introduces the adopted simulation environment and its associated parameters that are used for the implementation and validation of the proposed work; performance evaluation results and findings are also provided in this chapter;
- Chapter 7 concludes the thesis with improvements and future research directions.

Chapter 2 IT as a Service: An Overview

As outlined in the previous chapter, the objective of this thesis is to develop efficient decision strategies for migrating IT services to the cloud. This chapter establishes the necessary background to achieve this goal. It will first summarize the main concepts of the most adopted IT service model in literature. The chapter then proceeds to list the types of outsourcing costs associated with these cloud services. Outsourcing approaches and their limitation are also discussed in this chapter.

2.1 The Adopted IT Service model

In general, an IT service is a means of delivering value to staff or customers using IT resources. In this thesis, we will adopt the Information Technology Infrastructure Library (ITIL) definition of IT services given by Taylor et al. [14]: "A service provided to one or more customers by an IT service provider. An IT service is based on the use of Information Technology and supports the customer's business processes. An IT service is made up from a combination of people, processes, and technology and should be defined in a Service Level Agreement (SLA)."

One can further classify existing IT services into four main service layers as depicted by Figure 2.1. IT services at the lowest layer focus on networking services that connect the IT infrastructures. The services involved in managing these infrastructures are placed at the second layer. These include managing the local resources and platforms. Services in the third layer deal with ensuring the sufficient resources (e.g. storage, and computing resources) are available to the higher level services. Finally, these services at the highest level provide the direct interfaces for the customers and the staff.

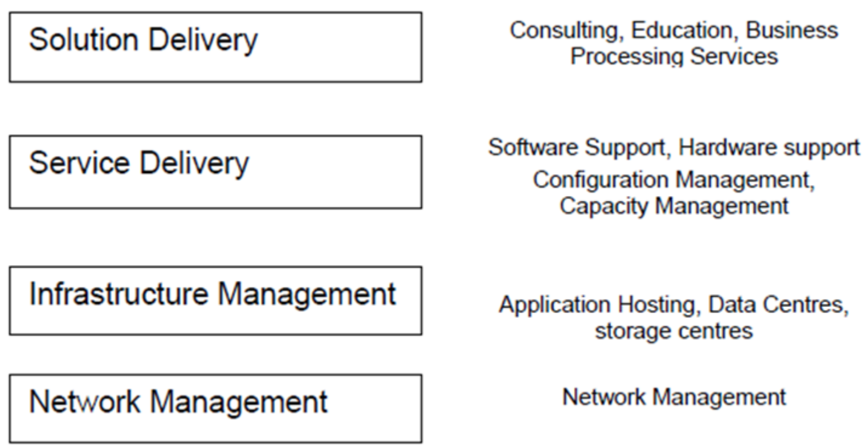


Figure 2.1: Layers of IT services [14]

The top-layer services introduced in Figure 2.1 deal directly with staff and customers or with various business processes.

2.2 IT Service Life Cycle

Service management “is a set of specialized organizational capabilities for providing value to customers in the form of services” [15]. As outlined in Figure 2.2, service management includes the following five phases:

1) Service Strategy

The Service Strategy phase provides guidance on how to design, develop and implement service management from an organizational capability perspective. It provides guidance on the principles underpinning the practice of service management which are useful for developing service management policies, guidelines and processes across the ITIL service lifecycle. Service Strategy covers these parts of IT systems:

- the development of markets, internal and external, service assets,
- the development of the service catalogue (a list of IT supported services in an enterprise), and
- the implementation of these strategies through the service lifecycle.

2) Service Design

The Service Design phase offers recommendations for the design and development of services and service management processes. It covers design principles and methods for converting strategic objectives into portfolios of services and service assets. The Service Design phase also guides organizations on how to develop design capabilities for service management.

3) Service Transition

The Service Transition phase provides recommendations for the development and improvement of capabilities for transitioning new and changed services into operations. This phase provides guidance on how the requirements of the Service Strategy phase are effectively realized in the Service Operation phase, while mitigating the risks of failure and disruption.

4) Service Operation

The Service Operation phase includes guidance on achieving effectiveness and efficiency in the delivery and support of services to ensure value for the customer and the service provider. Strategic objectives that were defined in the Service Strategy phase are realized through the Service Operation phase.

5) Continual Service Improvement

The Continual Service Improvement phase provides guidance in creating and maintaining value for customers through better design, introduction and operation of services. It combines principles, practices and methods from quality management, change management and capability improvement.

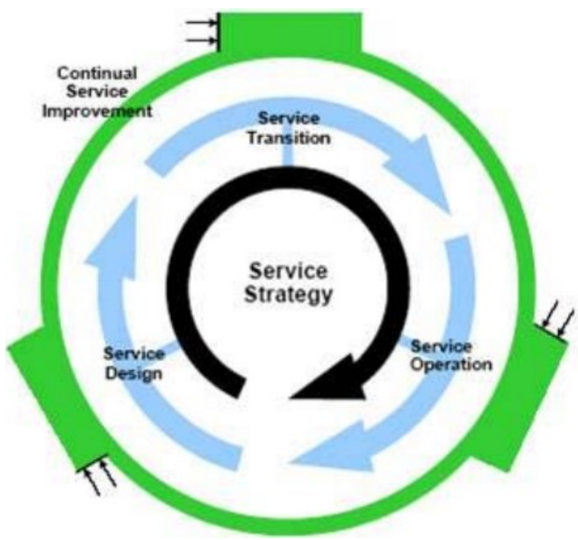


Figure 2.2: ITIL's Service Management [15]

Enterprises usually adopt the ITIL model because of the following benefits [16]:

- Alignment with business needs

  The ITIL is beneficial to enterprises because it provides guidance to IT organizations to proactively recommend solutions as a response to business needs. The ITIL Service Strategy phase provides the opportunity for enterprises to understand the business's current and future needs and develop service offerings that can address them [16].

- Negotiated achievable service levels

  Business and IT service providers become true partners when they can agree upon realistic service levels that deliver the necessary value at an acceptable cost [16].

- Predictable, consistent processes

  Customer expectations can be set and are easier to meet through the use of predictable processes that are consistently applied. Good-practice processes provide a solid foundation on which to lay the groundwork that are necessary to meet regulatory compliance requirements [16].

- Efficiency in service delivery

  Well-defined processes with clearly documented accountability for each activity can significantly increase efficiency. Service delivery tasks can be optimized by evaluating efficiency metrics, which indicate the time required to perform each activity [16].

- Measureable, improvable services and processes

  Service quality can be measured by analyzing key performance indicators (KPI). IT organizations can monitor their Service Level Agreements (SLA) by using proper metrics and making improvements as necessary. Consistent and repeatable processes can be measured and therefore can be better tuned for accurate delivery and overall effectiveness [16].

2.3 IT Service Dimensions

In an ITIL (IT Infrastructure Library) service management model, an enterprise has business strategies which govern the design of business processes' objectives. Through the Service Design ITIL phase, each business process is architected to use different IT services to deliver the process output [17]. Since IT services are at a lower level in the functional design view with a well-defined scope, parts of these services can be migrated to cloud services and hence managed using a service-oriented management model instead of relying on traditional application-oriented management models.

In recent years, there has been a shift from device centric and application-oriented management to service-oriented management model (IT Service Management). Because of this shift, IT service providers are facing newer challenges: customers are demanding a higher level of cost transparency and comparability with other service providers. As such, IT service providers focus on defining standardized and reusable service components and optimizing the internal delivery of IT services [18]. In order to understand the interdependencies of different services for different systems, a conceptual model is needed. The IT service model includes these dimensions:

- The customer dimension: this dimension provides a representation of the customer who is involved in the service delivery process. Since an IT service maps to business processes that they support within the customer organization, it is appropriate to model the customer business process. IT service model should represent not only the customer but also user needs [18]. In this dimension, user's needs are satisfied by using IT services. It is important to understand the user's needs rather than looking at their existing processes because during outsourcing, processes can change as long as user's needs are satisfied. In a later section of this thesis, it introduces that in-house processes can be migrated to clouds entirely using Software as a Service (SaaS).

- The output dimension: this dimension focuses on the service outcome that has an impact on the customer. Bundling services to customers represent the sales and marketing perspective of an IT service. One example is an e-mail service bundle which includes an e-mail client application plus software support (e.g. 24 hour technical support from e-mail vendor). IT services are more than just software and infrastructure, they may also include professional services which are human resources (staffs and consultants) performing technical work such as software configuration. One example of professional services is software configuration of an SAP (an Enterprise Resource Planning application) implementation; when a company decides to implement the SAP application, it often has a professional services contract with SAP to help customers migrate their business processes and to configure SAP [18].

- The process dimension: this dimension focuses on delivery processes for a service (e.g. provisioning processes) which are necessary activities to provide a service. The delivery processes for an IT service requires a high level of human involvement. First-level customer support is a good example of a manual service. As such, staffs constitute an expensive resource in the production of IT services. Since the activities related to delivering an IT service consume different resources and contribute differently to Quality of Service (QoS) and costs, it is useful to model them. As such, QoS procedures in the delivery process are also part of the overall process dimension [18].

- The resource dimension: this dimension describes staff and infrastructure that are essential to the delivery of these IT services. Technical resources include applications, hardware, and networks [18].

The conceptual IT service model is shown in Figure 2.3 using a Unified Modeling Language (UML) diagram. The customer dimension consists of a *contractor class* that represents a contractor as buyer of an IT product. An *IT Product* is a sales-oriented bundle of IT services. The relationship between the contractor and service provider is depicted through the associated contract class which contains an SLA. The *user class* represents the end user accessing IT services and the access point (such as a portal) to perform activities. In turn, these activities are a part of a business process that requires resources to deliver its output [18]. The IT service model is used by IT systems to deliver value to customers.

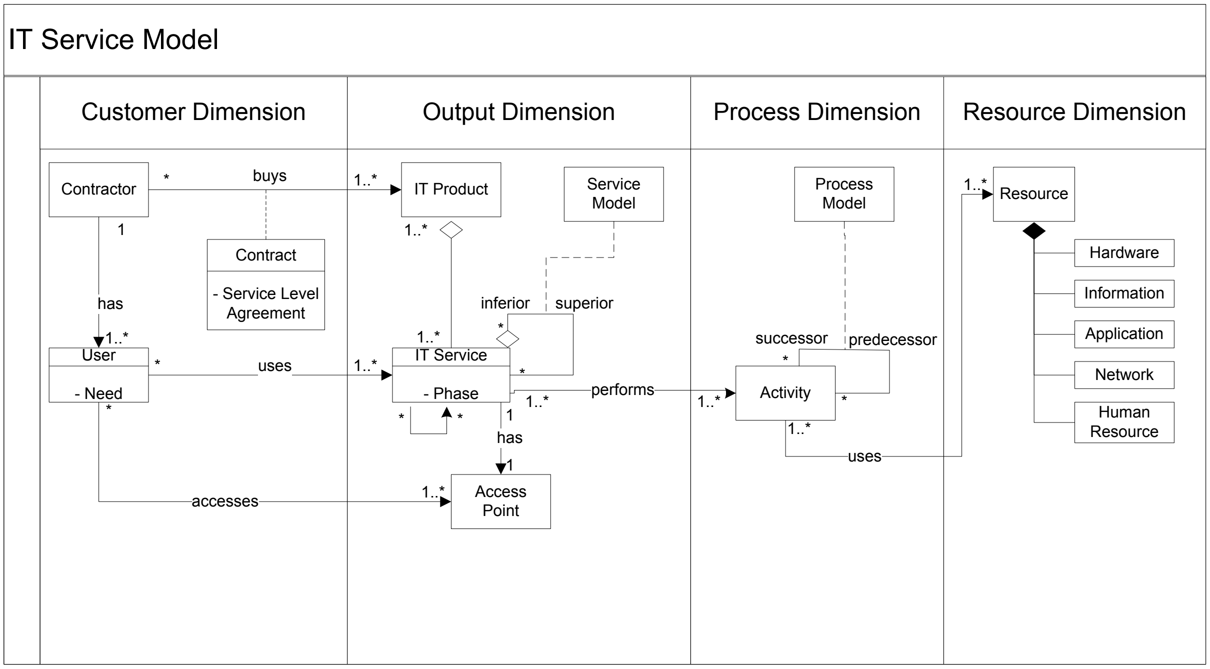


Figure 2.3: A Conceptual Model of IT Services [18,15]

Figure 2.3 consists of IT products in the "output dimension". The IT System is defined as a bundle of IT products that support business processes with information processing, provisioning and storage [17]. The elements of an IT system consists of: i) services, ii) operations, and iii) dependencies. In turn, a software service has components. Each operation has a number of configuration parameters. Finally, dependencies describe of a set of ordered operations that must be performed together. For example, if a software CRM service requires a database component, the database server installation and backup configuration should be included in the installation operation as a dependency. Mathematically, this dependency relationship can be visualized as a directed graph where operations are vertices and dependencies are edges between operations [19].

Figure 2.4 depicts a directed graph example that describes operation dependencies. In this figure, a CRM service ($service_1$) depends on a database operation ($operation_1$) which in turn also depends on a database server operation ($operation_2$) and a database backup ($operation_3$) operation.

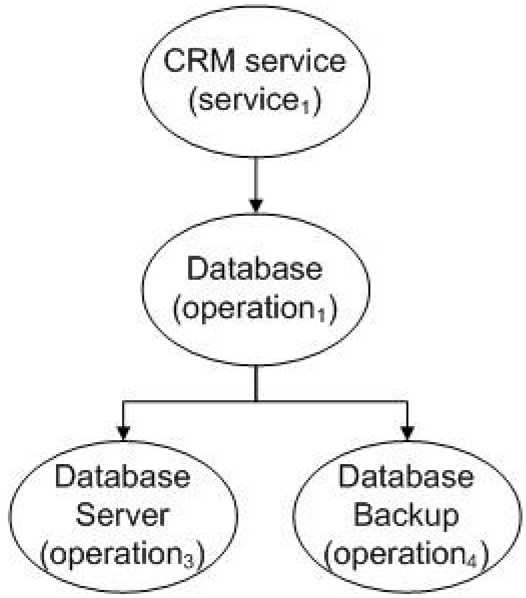


Figure 2.4: An example of Directed Dependency Graph [19]

2.4 From Business Objectives to Business Processes

Having described a model for IT services, the next step involves identifying the link between the high level enterprise objectives and these IT services.

Figure 2.5 shows that a business model represents the architectural level which acts as the idle layer or link is between business planning and IT implementation levels [20]. A business model is a conceptual and architectural implementation of a business strategy that describes the logic of a "business system" for creating value, which lies behind the actual processes. These three layers create what is known as the business logic triangle. The following is a description of the functionalities of these three layers.

- Conceptual (Planning) layer: business strategy which is the overall goals and objectives.
- Architectural layer: business model which contains a set of objects, concepts, and their relationships with the objective to express the business logic [21].

- Operational (Implementation) layer: business processes which contains sets of business logic to achieve the implementation of the business model.

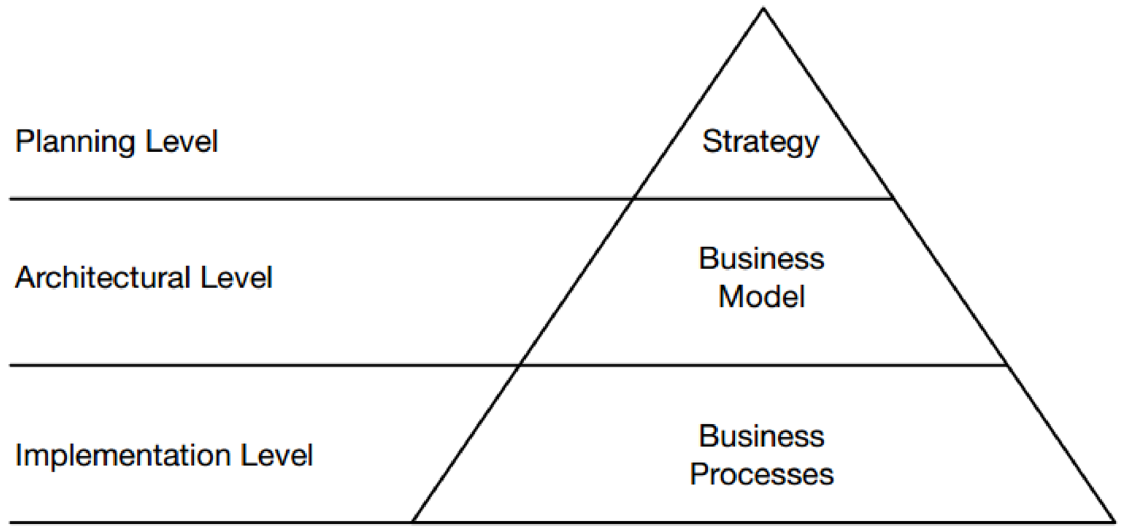


Figure 2.5: Business Logic Triangle [20]

According to Martens and Teuteberg [6], business processes are supported by IT services. Business processes could not be properly run without IT services. The authors have a "relative importance" formula for calculating the importance of a service to a business model using a dependency parameter as an input. This dependency parameter is derived from the IT architecture and the business process documentation [6]. The graphical representation of the relationship between business process and IT services is depicted below in Figure 2.6.

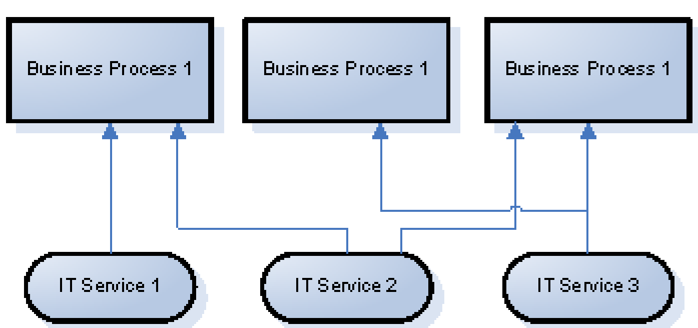


Figure 2.6: Business Process Mapping to IT Services

Having a proper mapping of business processes to IT services can determine the degree of interdependencies which aid the decision in deciding which set of IT services to migrate to cloud services together or none at all. When a set of IT services that are tightly coupled together, having them together either within the enterprise network or all on a single cloud vendor can reduce coordination cost.

2.5 IT Service Outsourcing and its Costs

As indicated before, outsourcing refers to services that were traditionally performed inside an enterprise being performed by external service providers. At a systems level, the broad definition of outsourcing is the practice of turning over part or all of an organization's information system (IS) functionalities to external service provider(s) [1].

Since each IT service has parameters that define the level of quality, when dealing with contractual obligations to quality of service, the ITIL recommends that service provision be formalized in a Service Level Agreement (SLA) [15]. Having a precise SLA is important when dealing with external vendors and especially when outsourcing services to the cloud.

An IT outsourcing decision is constrained by [50]:

1) The total cost of assigning N services.
2) The capacity constraint of each resource.
3) The cost due to the inter-dependence of two services.
4) The in-house versus outsourced costs of the same service.
5) A fixed cost.
6) The per transaction cost.
7) Coordination of information flow between services.
8) Unit cost of managing coordination information flow from different vendors.
9) Resource availability.

10) Amount of resources required.

The incurred fixed cost is comprised of the setup cost plus the coordination and monitoring costs for internal operations.

An IT outsourcing decision is also subject to resource limitations such as a finite number of employees that can perform the job in-house, therefore, a constraint on the cost model is that the demand exceeds the capacity. The term “unmet demand” is defined as the potential revenue that an enterprise could have made if it has more capacity. An enterprise can leverage cloud services to meet demands that exceed its capacity if the cost is justifiable. A decision function can aid enterprises to make this decision.

An outsourcing decision function uses the outsourcing fixed and variable costs to determine which services to outsource. At a high level, costs are categorized into three main areas: planning phase, operational phase, and decision phase. The planning phase represents costs incurred at the investigation stage of analyzing the cost benefits of outsourcing. The operational phase represents on-going cost from outsourcing [6]. At the operational phase, the decision model considers costs of setting up IT services with outsourced vendors; this phase also considers costs associated with running the services and losses that an enterprise will incur. At the decision phase costs are rolled up to three overall costs that the decision function will use: transaction costs, IT service costs, and expected loss. Transaction costs consist of costs incurred during the planning phase which are negotiation, adoption, and allocation costs, plus costs incurred during the operational phase which are maintenance, coordination and agency costs. IT service costs are associated with usage of the IT service. Lastly, expected loss consists of calculated risks associated with confidentiality, availability, and integrity loss. The following diagram depicts these costs and losses that are used by the decision function.

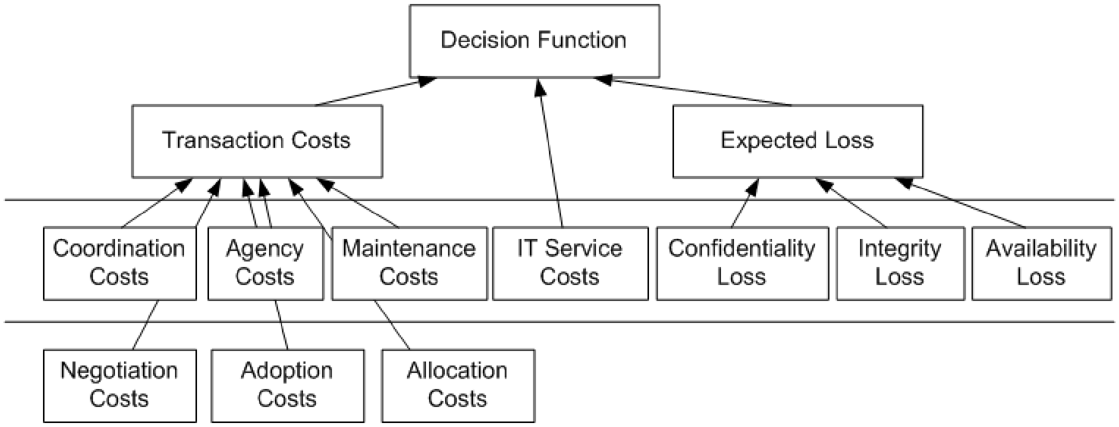


Figure 2.7: Activity-Based View of Decision Model [6]

The following list further explains the different types of costs and losses that the decision function uses:

<u>Planning Phase</u>

- Negotiation costs: associated with SLA definition
- Allocation costs: multi-sourcing with different providers
- Adoption costs: integrate an IT service into the user company's IT landscape

Operational Phase

- Maintenance costs: cost of running computers and data center
- Coordination costs: SLA management, provider management
- Agency costs: monitoring cost to measure performance data
- IT Service costs: usage dependent and usage independent elements
- Confidentiality, Availability and Integrity Losses: expenses arising from risk of technical issues and cost of mitigating those risks

Decision Phase

- Decision Function: activities arise necessary for decision-making and for procurement and implementation of Cloud Computing service. The cost of identifying the optimal service that achieves the right balance of risks and costs.
- Transaction Costs: is a sum of: negotiation costs + allocation costs + adoption costs + maintenance costs + coordination costs + agency costs.
- Expected Loss: expenses due to violation of a security objective; cost of switching to a provider and other efforts to make the service available again; loss in sales incurred during the downtime.

Costs associated with these three phases can be summarized as shown in Figure 2.7.

2.6 Approaches for Optimal Outsourcing

Enterprises have various criteria in deciding what services to outsource. The multiple criteria decision aiding (MCDA) method is used to evaluation a set of alternatives using a criteria list and a relationship rating scheme. The MCDA method requires an enterprise to construct a relationship matrix which consists of a list of criteria, a weight for each criteria, and activities to be outsourced. The output of a MCDA method is a ranking of activities to be potentially outsourced [22].

Some of the most dominant models of outsourcing are outlined below [22]:

The Single Vendor Model

The single vendor outsourcing model means that an enterprise does due diligences of selecting one vendor that fits best according to its outsourcing criteria. The main limitation of this strategy is that there are other vendors that can outperform the selected vendor in some of the criteria [22]; the single vendor model finds a vendor that fits best of all criteria on average. The Enterprise Resource Planning (ERP) implementation case study can illustrate the difference between a single-vendor model and a multi-vendor model. The Enterprise resource planning (ERP) software is a platform for supporting enterprise-wide business processes. The ERP software has multiple modules that each can use a different vendor. The advantage of a single-vendor solution is that it is generally easy to maintain as enterprises do not need to concern with integration with other modules from different vendors [23].

The Multi-Vendor Model

The "Best of Breed" IT strategy is another term to describe the use of multiple vendors to implement an IT system. Using the ERP implementation case study as an example, some enterprises use software from different vendors to implement different modules of their ERP system. The advantage of the multi-vendor model is that enterprises have the flexibility to select a vendor that fits better at a module level. The challenge of a multi-vendor model is that the enterprise needs to integrate these modules together [23]. This multi-vendor approach requires enterprises to put a significant effort to monitor and coordinate vendors. The advantage of this approach is that it allows better services and lower costs [22].

Integrated Supplier Model

The integrated supplier model offers advantages over the single vendor and multi-vendor models by improving the coordination problem. Enterprises that use the integrated-supplier model outsource their services to a main outsourcer that, in turn, manages different suppliers. The main outsourcer acts as a "go to" person and it is responsible for the performance of all suppliers. The advantage of this model is that it lowers the number of communication channel since an enterprise interacts with only the main outsourcer [22].

The IMP Group's Interaction Approach [24]

One approach for enterprises to manage IT outsourcing is to treat the relationship like a buyer and supplier interaction process. The interaction process was first introduced in the International Marketing and Purchasing (IMP) group's International Model. The model focuses on the factors which lead to closer relationships between the buyers and sellers. The interaction approach is based on inter-organizational theory and on transaction cost theory. This model focuses on both short-term and long-term relationships in buyer-supplier ventures. It views interactions between customers and suppliers as a series of short-term social interactions influenced by the long-term business process, reflecting the developmental aspects of relationships. The model delineates the exchange elements that may be regularized over time to improve cooperation between the parties [25].

Project based Outsourcing

Project-based outsourcing is an approach to outsource the entire project to an external company by defining the project requirements and advertising these requirements for IT-consulting companies to apply (bid for a project). Since every project has a planned start and end date, a project is temporary by definition; enterprises can manage costs during the defined project timeline without the burden of long-term staffing costs. Project-based outsourcing firms are firms that are set up around projects. These project-based outsourcing firms offer services such as integration of products or IT systems [26]. Within a pure project-based firm, projects "embody most, if not all, of the business functions normally carried out within departments of functional or matrix organizations" and act as the main mechanisms for coordinating and integrating them [27].

2.7 Limitation of Outsourcing

Outsourcing can occur at these levels: core, near-core, and non-core services [28]. The limitation of core and near-core outsourcing is that enterprises lose competitive edge because core outsourcing requires enterprises to transfer specialized knowledge necessary to outsourcers. Gilley and Rasheed [28] propose that outsourcing activities near their strategic core will achieve lower levels of performance relative to enterprises that retain tight control over these activities.

Although there are potential benefits to outsourcing, some argue that reliance on outside suppliers is likely to lead to a loss of overall market performance. One limitation of outsourcing is that the outsourcer will not take the initiative on innovation as they are executing the tasks according to how it is defined; the onus is on the enterprise to review their processes and inform the outsourcer for any changes.

A major limitation of outsourcing is communication with outsourcers. Communication gaps can occur at the following levels:

- Intra-team coordination
    - When members of the team are not centrally located, intra-team communication becomes difficult [29]. For cost saving and time-to-market strategies, it is possible that some members of the team are located in other countries in different time zones; as such, communication between team members is not instantaneous.
- Inter-team coordination
    - Communication gap among distributed teams is the root cause for lack of coordination. Improper coordination leads to delays in deadlines, missing functionalities and development of interdependent components also gets affected [29].

2.8 Summary

Business strategy planning drives business model design which in turn drives business process implementation. Business processes are then decomposed to different IT services for which a conceptual map can be developed. Once the dependencies of different IT services are determined, enterprises can use this information to calculate the cost of outsourcing services to the cloud as an alternative.

This chapter summarized the ITIL service model and its components. The IT service model is used as a standard for outsourcing and it consists of four dimensions: customer, output, process, and resource. An IT system is built to implement a set of business processes which use IT services to fulfill the business model. IT services can be outsourced by analyzing the three levels of the business logic model: planning, architectural, and implementation level. A decision model is used to determine mathematically whether to partially or fully outsource an IT service using transaction costs, IT service costs and expected loss as inputs. The next chapter provides an overview of the cloud computing paradigm and how enterprises can leverage the services that cloud vendors offer to lower IT expenditure.

Chapter 3 The Cloud Computing Paradigm – An Overview

According to the National Institute of Standards and Technology (NIST), "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [30]. Figure 3.1 below depicts a visual model of NIST's definition of Cloud Computing [31].

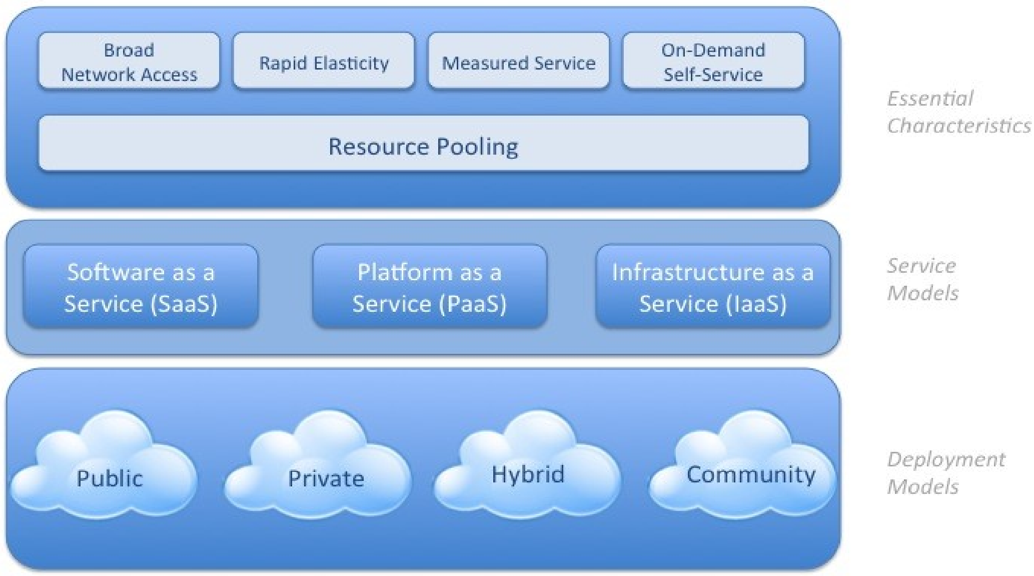


Figure 3.1: NIST's working definition of Cloud Computing [31]

The concept of cloud computing is based on an outsourcing model where IT services are procured from service providers and accessed through the Internet. An example of cloud computing service is an ERP CRM module that is hosted at a cloud vendor site (e.g. Salesforce.com [32]) where servers and web application are managed on the Internet by the cloud service provider.

Cloud computing can meet software applications, storage and processing needs and payment is made only for the resources that are used. The players in a cloud model are the cloud service providers (third parties, suppliers or brokers) and cloud consumers (individual end-users, enterprises, or IT staff). Services are governed by contractual service level agreements (SLA) that specify consumer requirements and the provider's commitment [33].

The idea of Cloud Computing is sharing of computing resources. The concept of sharing resources is not new; early mainframe computers allowed multiple users to share physical access and CPU time from multiple terminals to get better returns from their investments since mainframe computers are expensive. In the 1950s, scientist Herb Grosch envisioned that the world would operate on dumb terminals powered by about 15 large data centers [34]. There has been several debates over the evolution of cloud computing. While some people relate cloud computing to grid computing, others believe that it is an extension of grid computing. Grid computing is “a system that coordinates resources which are not subject to centralized control, using standard, open, general-purpose protocols and interfaces to deliver non-trivial qualities of service” [35]. One major differentiator is that in grid computing, high utilization is achieved through the allocation of multiple servers to a single task; on the other hand, in cloud computing, high utilization is achieved by allowing one server to compute several tasks concurrently through virtualization [35].

There are two viewpoints regarding cloud computing novelty. One view sees cloud computing as a new technology because it changes how applications and infrastructures are invented, developed, deployed, scaled, update, maintained, and paid for according to pricing model. On the other hand, the other view claims that cloud computing is not a novel concept because it is based on established best practices. Cloud computing incorporates virtualization, on-demand deployment, Internet delivery of services, and open source software [36]. Foundational elements of cloud computing includes primary technologies such as virtualization, grid computing, service oriented architectures, distributed computing, broadband network, browser as a platform, free and open source software.

The cloud model is composed of five essential characteristics, three service models, and four deployment models which are detailed in the following sections.

3.1 Essential Characteristics of Cloud Computing

The five essential characteristics of cloud computing are listed below [12]:

1) On-demand Self Service

Consumers are given access to computing capabilities such as network storage and server time automatically as needed without requiring interaction with service providers.

2) Broad Network Access

Computing capabilities are provisioned through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g. mobile phones, laptops, and PDAs).

3) Resource Pooling

Service Providers pool computing resources to serve multiple consumers using a multi-tenant model with different physical and virtual resources, which are dynamically assigned and reassigned based on consumer demand. Consumers do not have control over the location of the provided resources except at a higher level of abstraction (e.g. country, state, or data center). Examples of provided resources include storage, processing, memory, network bandwidth, and virtual machines.

4) Rapid Elasticity

Computing capabilities are provided rapidly and elastically that can be automated with even triggers. The benefit of rapid elasticity are that when there is a peak of demands for an IT service, computing capacities can quickly scale out and when demand is back to normal, additional capacities are released to quickly scale in. The computing capabilities to the consumer are unlimited and can be purchased at any time.

5) Measured Service

Resource usages are automatically controlled and optimized in cloud computing systems by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, and bandwidth). Resource usage can be monitored, controlled, and reported to provide transparency for parties of the consumed service.

These characteristics of cloud computing are the foundation of the three cloud computing service models.

3.2 Cloud Computing Service Models

Cloud vendors provide different functions giving consumers control over the type of cloud services based on their business needs. According to Mell and Grance, cloud computing has three service models [30]:

<u>Software as a Service (SaaS):</u>

SaaS refers to using a cloud software application within an organization. The target customers at the SaaS layer are users that run applications from SaaS providers in the cloud. One example of SaaS hosted on a public cloud is Saleforce.com that provides Customer Relationship Management (CRM).

<u>Platform as a Service (PaaS):</u>

PaaS consists of programming languages and tools to deploy onto the cloud infrastructure. The target customers are SaaS users that are intended to deploy applications into the cloud. One example of PaaS is Microsoft Azure. PaaS provider offers services to the user to deploy their application solution without the need for them to maintain the underlying hardware and platform software; an abstraction from the infrastructure layer.

<u>Infrastructure as a Service (IaaS):</u>

IaaS refers to the processing, storage, networks, and other fundamental computing resources. The target customers are application providers. IaaS providers offer a configurable set of core infrastructures to their customers that includes servers (processing), disk space (storage and backup), and connectivity (networks). In a multi-tenant concept, even though other IaaS customers are using the same servers, the access is isolated to just the area (virtual machine) that is defined by the provider; this means that even though there are multiple tenants hosted on a single server, each tenant cannot access someone else's virtual machine.

Although each of the cloud computing service models has the five essential characteristics, they differ in the level of control for their end users.

3.3 Level of Control

The aforementioned three types of service models differ from one another in the amount of control given. Figure 3.2 depicts the values at each level; IaaS offerings can create a foundation for PaaS offerings, and PaaS offerings can be a foundation for SaaS. On the right side of the figure, it shows the targeted customers at each services; PaaS are mainly targeted for network architects that can install their own platforms; PaaS are targeted for application developers that use the established platform provided by the provider to deploy their solution; and lastly SaaS are for end users that use the software solution [37].

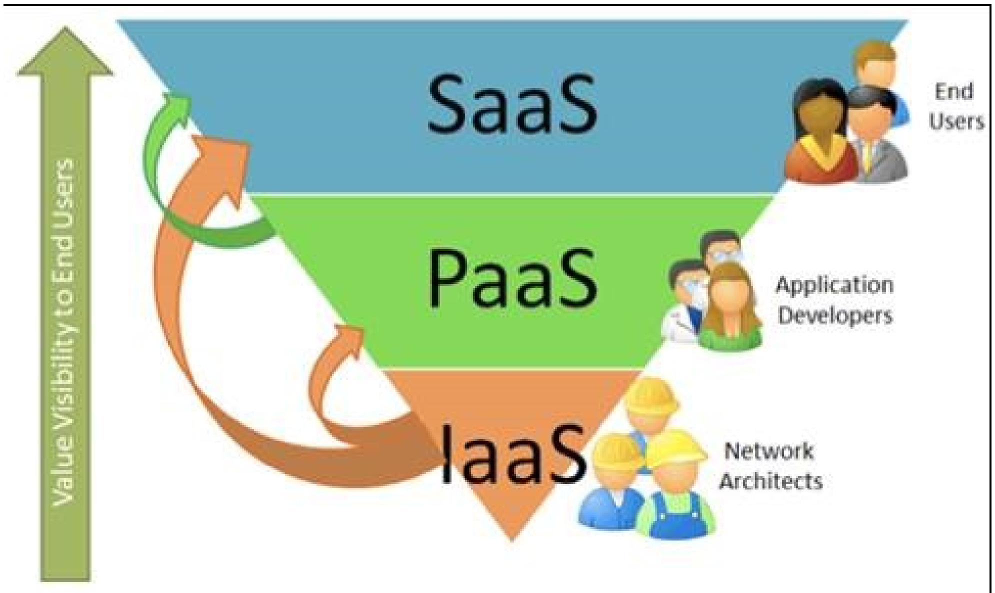


Figure 3.2: Cloud Computing Stack [37]

3.3.1 Differences between Service Models

In SaaS, consumers run the applications from a SaaS provider on a cloud infrastructure, which are accessible from various client devices through a thin client interface such as a web browser. Also, consumers do not manage or control the underlying cloud infrastructure which includes network, servers, operating systems, storage, and application capabilities, with the possible exception of limited user-specific application configuration settings [12]. There are two distinct modes of SaaS: simple multi-tenancy and fine grain multi-tenancy [38]. Examples of SaaS applications include Salesforce.com, Google's Gmail and Apps, instant messaging from AOL, Yahoo and Google, and Voice-over Internet Protocol (VoIP) from Vonage and Skype [39]. At this layer, the consumers do not need to worry about backup of data, disaster recovery planning of server failure, and infrastructure maintenance as they are all managed by the SaaS provider.

In PaaS, consumers have access to the cloud infrastructure, consumer-created or acquired applications created, using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly the application hosting environment configurations. PaaS services include application design, development, testing, deployment, and hosting. Other services include team collaboration, web service integration, database integration, security, scalability, storage, state management, and versioning [12]. Examples of PaaS include Microsoft's Azure and Salesforce's Force.com. Another example is the Appscale – Google App Engine (GAE), a cloud-computing platform for developing and hosting web applications in Google-managed data centers [39]. Consumers of PaaS are developing their own software using the APIs and platform provided by the PaaS provider. Whereas for SaaS, the provider builds and maintain the software which has been tested by the providers themselves; consumer of SaaS are pure users of the application without understanding of how the software is built. Consumers of PaaS, similar to SaaS, also do not need to worry about backup of data, disaster recovery planning of server failure, and infrastructure maintenance as they are all managed by the PaaS provider.

In IaaS, consumers are provided with computing capabilities such as storage, networks, and other fundamental computing resources to deploy and run arbitrary software. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications, with possible limited control of select networking components (e.g., host firewalls) [12]. Examples of IaaS are Amazon's Elastic Compute Cloud (EC2) and Simple Storage Service (S3) [39]. IaaS is different compared to SaaS since IaaS consumers are deploying their software and they are solely responsible for the business logics/rules, software maintenance, and quality (testing of their application). IaaS is different compared to PaaS since IaaS consumers install, maintain, and decide on their own platform, IaaS consumers have full control over the development environment and how a software application is deployed. PaaS consumers are restricted to the environment that the provider has setup and that PaaS consumers have to adapt to that platform; for example, a PaaS provider hosts a Websphere server which only J2EE web applications can be deployed, as such, .NET applications are not compatible with this platform. This differs from IaaS since the IaaS consumers install their own platform directly on the server. At all three levels, consumers of SaaS, PaaS, and IaaS do not need to worry about backup of data, disaster recovery planning of server failure, and infrastructure maintenance as they are all managed by the underlying IaaS provider.

Figure 3.3 [40] shows the different area that a cloud consumer is responsible for in each of the three services:

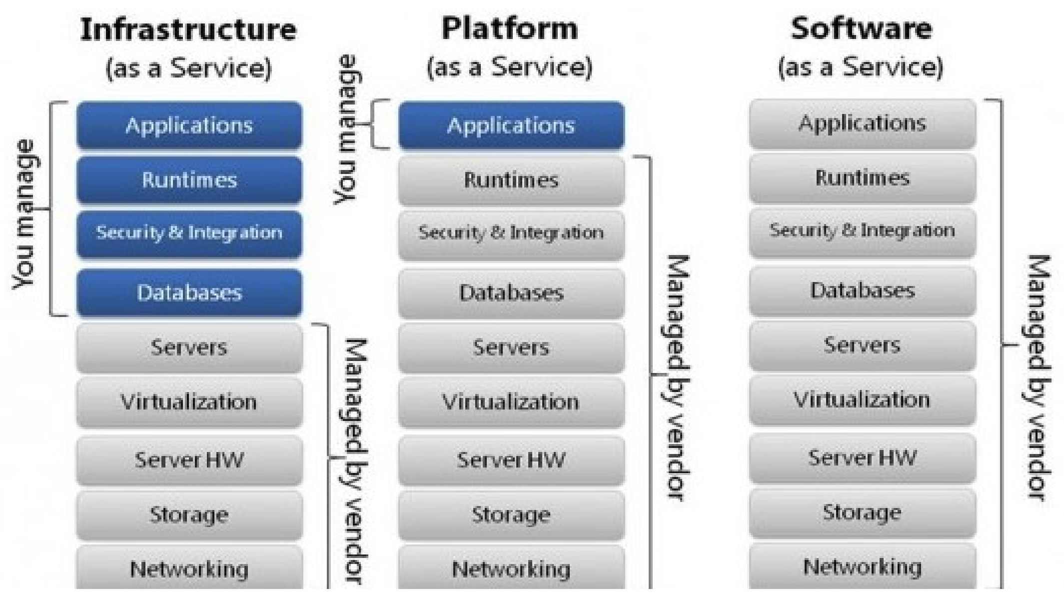


Figure 3.3: Level of Responsibility between Cloud Service Models [40]

At a conceptual level cloud computing offers three service models for which each model has different level of control for its end users. At an implementation level, cloud services can be deployed internally within an enterprise's data center (private cloud), in a shared group of enterprises (community cloud), externally (public cloud) through a cloud vendor, or a mix of both internal and external (hybrid cloud). The next section introduces these four deployment models in cloud computing.

3.4 Deployment Models in Cloud Computing

The deployment models in cloud computing defines the ownership and control of architectural design and the degree of available customization. According to Mell and Grance, cloud computing has these four deployment models [30]:

3.4.1 Private Cloud

In a private cloud deployment model, organizations have their own Data Centers (infrastructure) within the company with in-house employees or contractors (often refer to as insourced). One motivation of the private cloud model is data security; sensitive data is kept and controlled locally within an organization. This type of cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g. business units). It may be owned, managed, and operated by the organization, a third party, or some combination and also, may exist on or off premises.

3.4.2 Community Cloud

In a community cloud deployment model, infrastructures are shared by several organizations, as such, creating a community. A specific community of consumers from organizations that have shared concerns provides this type of the cloud infrastructure for exclusive use. For example, mission security requirements, policy, and compliance considerations. It may be owned, managed, and operated by one or more of the organizations in the community, a third party or combination, and may exist on or off premises.

3.4.3 Public Cloud

In a public cloud deployment model, cloud services are made available to the general public. One motivation of the public cloud deployment model is flexibility in pricing and economy of scale. Public cloud vendors offer low setup cost and pay-per-use or pay-per-user pricing model which in some situation such as start-up companies, it may be beneficial. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises and fully controlled by the cloud provider.

3.4.4 Hybrid Cloud

A Hybrid cloud deployment model is a mix of Private, Community and Public cloud networks. This deployment model offers the organization the flexibility and a balance between privacy, performance, and cost savings. For example, sensitive data can be kept local to the enterprise while less sensitive data can be migrated to the cloud [41].

Once enterprises determine at which cloud service models and at which deployment models to migrate their IT services, it is important to determine the cost associated with cloud vendors. The next section introduces different cloud pricing models.

3.5 Pricing Models in Clouds

One of major advantages of using Cloud Computing is that there is no large upfront cost for infrastructure. As such, the pricing model for Cloud Computing is attractive for cash strapped startup companies. Pricing is one of the most critical decisions that an enterprises make when planning for an introduction of a new IT service [42]. There are three categories for price structure for Cloud vendors:

1. Fixed Pricing
    - Pay-Per-Use: Customers pay in function of the time or quantity they consume of a specific service.
    - Subscription: Customers pay a flat fee in order to access the use of a product or to profit from a service.
    - List price/menu price: A fixed price that is often found in a list or catalog [43].

    One example of fixed pricing is Amazon EC2 where usage is charged per hour on a pay-per-use model. Customers also have the option of a subscription model which Amazon calls "Reserved Instances" where the subscription will have the benefit of a lower hourly charge per instance [44].

2. Differential Pricing
    - Service feature dependent: Price is set according to service configuration. Includes bundling of different services.
    - Customer characteristic dependant: Price is tailored to the characteristics of every single customer.
    - Volume dependant: Differentiates prices on the basis of purchased volumes.
    - Value-based: The final price will strongly depend on the customer's valuation of a value proposition [43].

    One example of differential pricing is CloudOne which offers cloud solution tailored to IBM Rational tools. The price depends on which IBM services are requested and a cloud environment is built and configured according to customers' needs.

3. Market Pricing
    - Bargaining: The price outcome depends on the existing power relationships between the parties involved.
    - Yield management: the best pricing policy for optimizing profit is calculated based on real-time modeling and forecasting of demand behavior.

- Auction: Price is set as buyers bid in increasing increment of price. Reverse auction: price is set as sellers bid in decreasing decrements of price.
- Dynamic market: Price is the outcome of a large number of buyers and sellers that have indicated their price preference, but are not able to influence this price as individual sellers [43].

One example of market pricing model is Amazon EC2 Spot Instance that allows customer to name their own price to bid at market price which is based on demand and supply. Through this method, the VM instance will be launched if the bid price is higher than the market price and is committed on an hourly basis. However, if the market price rises above the bid price in the following hour, the VM instance can be terminated (if there is shortage of resources). The advantage of an auction market pricing model is that it can lower cost for time-flexible and interruption-tolerant tasks [45].

3.6 Summary

In this chapter, we have summarized the main concepts of the cloud computing paradigm. We have shown that cloud computing may offer three different services: Software as a Service (SasS), Platform as a Service (PaaS), and Infrastructure (IaaS). SaaS is outsourcing at the business process level. PaaS is outsourcing the deployment process of a specific platform (such as Java or .NET services). IaaS is outsourcing at the infrastructure level. Cloud computing offers different deployment model for enterprise to decide on the level of control; in a private cloud service model, the enterprise has full control of the hardware and staff operating the data center, whereas on the public cloud service model, the enterprise relies heavily on the cloud vendor; the hybrid cloud service model offers a mix of public and private models where sensitive information is stored in a private cloud and other information is stored in a public cloud; the community cloud service model offers a set of enterprises the right to use its cloud services.

Since cloud computing is a form of outsourcing, the IT service model is used as a guideline for mapping the different components of business processes. The next section of this thesis outlines differences between IT outsourcing and migrating services to the cloud by means of literature review.

Chapter 4 Literature Review – Outsourcing and Cloud Computing

This chapter summarizes findings from the literature relating to similarities and differences of outsourcing and cloud computing. Although cloud computing is a subset of generic outsourcing, there are specific costs that are incurred during cloud migration; these costs are discussed in this chapter. Cloud migration strategies are also researched by means of decision frameworks and models.

4.1 Why Clouds are better than Outsourcing

As described before, for enterprises looking to take the complexity out of managing their IT infrastructure, outsourcing has been a popular option prior to cloud computing because it allows them to hand over the problem to a professional-services companies that specialize in dealing with technology. Cloud migration is similar to outsourcing since it has the ability to simplify the complexity of managing IT infrastructure both at the level of the application and from a budget point of view [46]. Even though cloud migration strategies are less mature than outsourcing models, the former seems more promising since it involves getting the business to rethink at the strategic level of business needs and search for cloud vendors to transform their existing in-house processes using SaaS processes to lower costs and improve efficiency.

With the understanding that cloud computing can simplify the complexity of managing IT resources, it is important for businesses to have a strategy to decide what IT services to migrate to the cloud. The next section elaborates the differences between generic IT outsourcing and cloud outsourcing, and costs associated with cloud migration.

4.2 Outsourcing versus Migration to cloud resources

Outsourcing is a generic term of shifting in-house production to external organizations. The definition of IT outsourcing is an act of delegating or transferring some or all of the information technology related decision making rights, business processes, internal activities, and services to external providers. These external providers will then develop, manage, and administer these activities in accordance with the agreed upon deliverables, performance standards, and outputs, as set forth in the contractual agreement. The cloud computing concept is to transfer some or all IT operations required to run a business to the cloud which provides a flexible and a highly scalable technology platform for an organization's business operations [47]. Furthermore, a cloud migration strategy differs from an outsourcing strategy at the cloud service model. At the Software as a Service (SaaS) cloud service model, the vendor provides a fully supported application; as such, a new business process is derived. With a SaaS cloud migration, enterprises do not require internal resources at the infrastructure layer (such as server maintenances) because it is offloaded to a SaaS cloud vendor. One major difference between the cloud computing and outsourcing concept is that cloud providers grant users the right to use their software, databases, and an infrastructure; this is not the case in most outsourcing [48].

4.2.1 Service Pricing

The cloud computing model needs to take the cloud pricing model into account when deciding on a vendor [43], whereas the generic outsourcing concept is based on service contracts. The following text outlines the differences between different types of outsourcing contracts and cloud computing pricing models:

IT Outsourcing Contract Types:

1) Fixed Price (FP) Contract: fixed price, firm fixed price, fixed price with incentive fee, fixed price with economic price adjustment, purchase order [49].

2) Cost Reimbursement (CR) contract: cost plus fixed fee, cost plus percentage of cost, cost plus incentive fee, cost plus award fee, cost contract [49].

3) Time and Material (T&M) contract: hybrid of fixed price and cost reimbursement contract [49].

Cloud Computing Pricing Types:

1) Fixed pricing: Pay per use, subscription, list price/menu price [43].

2) Differential pricing: Service feature dependant, Customer characteristic dependant, volume dependant, value-based [43].

3) Market pricing: Bargaining, yield management, auction, reverse auction, dynamic market [43].

Even though there are similarities in generic outsourcing and cloud computing, the latter operates differently with the concept of "renting" a service from a vendor. On the other hand, IT outsourcing is like hiring a management company to manage buildings. One significant benefit of cloud computing is that there is no up-front cost to get your services up and running [47]; at a SaaS layer, since the software is already developed by SaaS vendors and available on their servers, consumers of SaaS services can "rent" this service without first developing the application; SaaS consumers will need to configure the SaaS application to incorporate into their business processes. The following table summarizes the differences between IT outsourcing and cloud outsourcing:

Table 1: Differences between IT Outsourcing and Cloud Outsourcing [47]

IT Outsourcing	**Cloud Outsourcing**
Outsourcing means transferring some or all of the IT related decision making rights, business processes, internal activities, and services to external providers.	The framework allows enterprises to manage the building blocks of IT services but without the challenges of complex IT architecture. Most of the benefits of the outsourcing are achieved without the majority of drawbacks. Enterprises retain some control of their business processes by customizing the cloud software.
Initial up-front cost required in most cases.	No up-front costs: the capital expense and installation are absorbed into the rental charges.
Services are not necessarily on demand.	On demand: near-instant scaling /adding of resources.
More capacity planning required for future demand.	Flexibility with regards to increasing/decreasing resources.
Lots of hidden costs.	Cost is more transparent.
High level of customization possible.	Less customization.
More project management, onshore and offshore coordination and governance	Less project management, onshore and offshore coordination and governance

required.	required.
Outsourcing vendors can offer value-added services along with strategic and management consulting.	Strategic and management consulting are beyond the scope of cloud computing.
Usually specify where data resides, how it is protected, who has access, which measures are in place to accommodate political or natural disasters.	Lack of provisions for compliance, business continuity, security, and privacy of data.
Any custom application development occurs on the top of pre-selected cloud platforms.	Uses a pre-determined set of cloud applications and infrastructure to meet IT demands.
Longer contracts.	Shorter contracts.

Cloud migration is a type of outsourcing subject to service inter-dependency. For cloud migration, the setup cost for outsourcing needs to consider the cost of transfer data from an in-house system to the vendor system.

Since cloud computing intersects with outsourcing, literature review related to outsourcing was analyzed. Outsourcing focuses on reducing fixed cost, transaction cost, and interrelationship cost subject to resource availability and capacity [50].

For literatures relating to cloud computing, they suggests that cloud computing is similar to make-or-buy decision where a service is partially outsourced. One major decision factor of using cloud computing is the small upfront cost; this is an important factor for cash-strapped companies with cash flow concerns. As such, the adoption rate for cloud computing is higher among smaller companies. The decision model from Martens and Teteberg [6] consider costs, risks associated with vendors, security and quality issues. They suggest that multi-sourcing can reduce cloud computing risks but it also increase coordination cost. It is difficult to measure the quality of a cloud vendor; enterprises can use a performance matrix that was evaluated by a third party to compare quality measurement of different cloud computing vendors [50].

Although minimizing cost is an important factor, maximizing revenue is also as important. In literature review relating to IPTV, the REVMAX (revenue maximization) algorithm maximizes revenue by accepting requests that generates more revenue when there is a limited resource to serve competing requests [51]. If the ultimate goal is to maximize profit, analyzing the revenue that would have gained by servicing one request versus another is important in a decision model. An IPTV example is the decision to service a paid service such as videos from Pay-Per-View channel versus a normal video stream that does not generate revenue when the network is over capacitated. Using the REVMAX approach, where there is limited capacity in IPTV delivery system, it justifies the decision to deliver a video stream to a service that generates higher revenue by not servicing other requested services even if there is a small cost of not meeting that SLA.

An example that summarizes the distinction of revenue based and profit based is described below.

<u>Premise of Example:</u>

There is capacity limitation in IPTV content delivery system and the system has to decide which service to delivery.

Revenue for Service A = $100

Revenue for Service B = $101

Cost of not meeting SLA for Service A=$50

Cost of not meeting SLA for Service B=$0

Table 2: Revenue based Decision Model vs. Profit Based Decision Model

Revenue Based Decision Model	Profit Based Decision Model
In Revenue based decision model, cost is not taken into consideration. Since Revenue for Service B is higher than Service A, **Service B is selected**. Revenue=$101	In a brute forced method, different scenario are analyzed: Scenario 1: To serve Service A Revenue=$100 Cost=$0 **Profit=$100** Scenario 2: To serve Service B Revenue=$101 Cost=$50 **Profit=$51** Service A generates a higher profit than Service B, **Service A is selected**.

Revenue based versus profit based calculation is an important distinction because it leads to the discussion of outsourcing cost as it involves multiple parameters in its decisions.

The cloud computing model differs from the normal IT outsourcing in the area where costs are associated with these categories [52]:

1) Training or Learning Curve
   - Training on cloud platform, offerings and technologies.
   - Identifying compatibility issues.
   - Training on third party tools.

2) Installation and Configuration
   - Installing and setting up environment in IaaS cloud, third party tool.

- Installing and setting up development tools.

3) Code Modification
   - Installing and setting up database connection to cloud database server.
   - Modifying source code for cloud compatibility issues.

4) Migration
   - Preparing the database for migration.
   - Migrating the local database to cloud database.
   - Preparing system for migration.
   - Migrating the application.

5) Testing
   - Writing test cases and testing the functionality of the application in cloud.
   - Performing test to validate local system works with database in cloud.
   - Performing test to validate system in cloud works with database in cloud.

Tran et al. [52] article defines tasks associated with the five categories are similar to Martens et al. [6] article for cost incurred during the planning phase and operational phase. The following table maps the tasks and phases between the two articles:

Table 3: Mapping of cost category to time definition

Cost Categories defined in Tran et al. [52] article	**Time Definition defined in Martens and Teuteberg [6] article**
Training or Learning Curve	Planning Phase - Adoption Cost
Installation and Configuration	Planning Phase - Adoption Cost
Code Modification	Operational Phase - Maintenance Cost
Migration	Planning Phase - Adoption Cost
Testing	Planning Phase - Adoption Cost

The next section of this thesis maps costs to different phases in a cloud decision framework.

4.3 Existing Cloud Migration Decision Frameworks

A cloud-decision framework is a set of guidelines that enterprises can follow to help them analyze alternatives using cloud solution. Figure 4.1 depicts Gartner's [53] cloud-decision framework which consists of the following main steps:

- Step 1: Prework
    - Create cloud computing core team
    - Define business objectives
    - Scope the effort
    - Establish cloud adoption principles
- Step 2: Business and application assessment
- Step 3: Vendor Selection Process
- Step 4: Mitigate Risk and Liability
- Step 5: Steady State

The activities defined in the Gartner [53] Cloud Decision model "Step 1: Prework" phase matches the Hajjat et al. [41] business logic triangle model at the "Planning Level" phase where business strategies are defined. Once business strategies are defined, "Step 2: Business and application assessment" maps to two levels of the business logic triangle model for defining the business model and business processes associated with the business model. Costs associated with activities defined in "Step 2: Business and Application Assessment" maps to the planning phase as described in Martens and Teuteberg [6] article; these costs are negotiation cost, allocation cost, and adaption cost. Costs associated with "Step 3: Vendor Selection Process" maps to allocation costs in the planning phase in Martens et al. [6] article. The activities in "Step 4: Mitigate Risk and Liability" maps to the decision phase in Martens et al. [6] article where a monetary value is assigned to expected loss such as confidentiality loss, availability loss, and integrity loss. In this phase, the transaction costs, expected losses, and IT services costs are summed up and used by a decision function to determine the optimal decision. Note that risks are represented in currency value based on expected losses in the Martens et al. [6] article. In the last step, "Step 5: Steady State", it maps to the operational phase in Marten et al. [6] article where costs associated are maintenance, coordination, and agency cost.

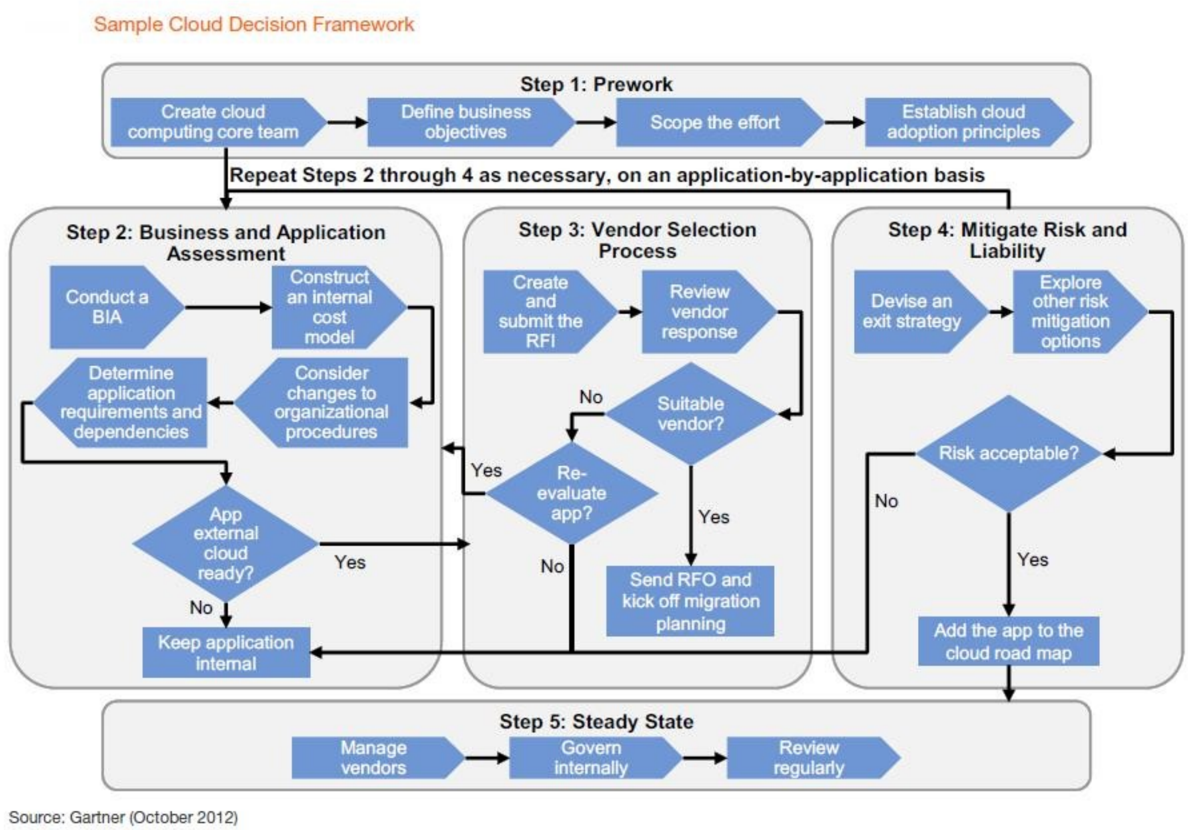


Figure 4.1: A Cloud Decision Framework [53]

The next chapter of this thesis proposes a framework that consists of a process to construct an internal cost model as illustrated in one of the box in “Step 2” in Figure 4.1. The analysis of internal cost model is necessary to determine the percentage of in-house versus outsourced services that can optimize profit. Outsourcing is then further decomposed into different dimensions which are described in The Cloud Cube Model by Jericho Forum. The purpose of the Cloud Cube Model (CCM) is to enable secure collaboration in the appropriate cloud formations [33]. This model has these dimensions to modeling cloud computing:

Dimension 1: External versus Internal

- In this dimension, it refers to the type of business model whether cloud services are externally or internally provided. Internal means private clouds and external means public clouds. This dimension is used as a basis of the proposed optimization model of this thesis where an optimal solution can be formulated with both external (cloud) and internal IT services to meet the demand.

Dimension 2: Proprietary versus Open

- In this dimension, it refers to how the cloud service is provided. Proprietary means paid services or contractors. Open means services rendered through open-source services or solutions or system or platform that allows sharing and free accessing of APIs. This dimension is relevant to the cloud at the platform as a service (PaaS) layer where some APIs are offered for free. One example on an open service is the Google Map API. The Google Map API is a free service that lets users embed Google Maps into their website of up to 2,500 requests per day [54]. The Google Map API for Business edition is an example of a proprietary paid service that has more functionality.

Dimension 3: Perimeterized versus De-Perimeterized

- In this dimension, it refers to architecture. Perimeterized in the context of cloud computing means infrastructure as a service (IaaS) and platform as a service (PaaS) or services, contracts, and supports using infrastructure and platform. Deperimeterized refers to Software as a Service (SaaS) or services that supports for software/application not restricted by hardware. One example of a perimeterized service is a customer relationship management (CRM) application that is deployed on an IaaS cloud. An enterprise may choose to continue its existing CRM processes by migrating its existing application to the cloud at the infrastructure level. Alternatively an enterprise can also migrate to the cloud using a deperimeterized SaaS service such as Salesforce.com which is a migration from their existing in-house CRM process to a new process on the Saleforce.com cloud.

Dimension 4: Outsourced versus Insourced

- In this dimension, it refers to the resources providing the cloud services. Insourced means having staff, equipment, and/or software within the company; when it is insourced, the company has a higher level of control. Outsourced means external to the company by means of purchased services (contracts). The insourced dimension refers to enterprises that implement their own private cloud with their own internal staff whereas the outsourced dimension is to hire external company manages their private cloud.

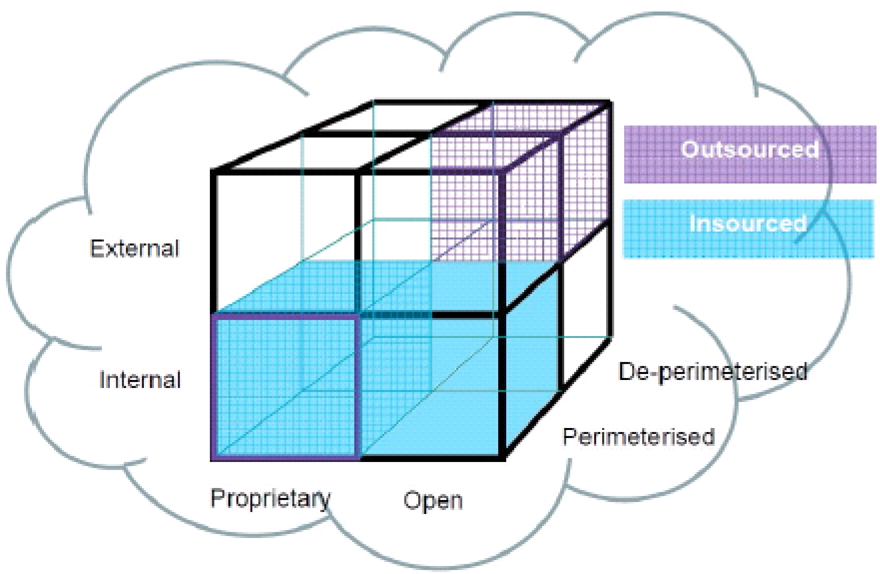


Figure 4.2: Cloud Cube Model [33]

For enterprises to be resilient in different network situations, cloud services need to be tested in different scenarios. Litoiu et al. (2010) suggest that testing the model would involve optimization at different real-life scenarios [55]:

Scenario: Burst of Load

SaaS layer: Application needs to identify additional capacity or load balancing.

PaaS layer: Find optimal decision to provide resources or request more from the IaaS by deploying and allocating additional VM.

IaaS layer: Managing storage, memory, and hardware for additional VM requests for on-demand request.

Scenario: Node Failure

IaaS layer: Ability to monitor failure and replace lost node by deploying new VMs and copying data.

PaaS layer: Respond by deploying replica containers.

SaaS layer: Deploy replica software.

Although cost is an important factor in outsourcing decision, there are risks associated with cloud computing.

4.4 Risks associated with Cloud Computing

The different types of risks associated with cloud computing are [6]:

Loss due to the violation of confidentiality

Sensitive information disclosed due to lack of control over provider's IT infrastructure

Loss due to the violation of integrity

Loss or manipulation of data in provider's data center

Loss due to the violation of availability

Response time and uptime of server which correlate to quality issue of cloud computing services

4.4.1 Data Confidentiality

Data confidentiality is a concern for public cloud computing because data and processing are done on a remote vendor's facility; one fundamental difference regarding data integrity in cloud computing is that the data owner has diminished control over its data.

The disadvantages of cloud computing includes [48]:

- inability to negotiate equitable terms (especially for public clouds);
- loss of control over data with concomitant potential privacy problems;
- application software that may be less than the optimum for a particular user's situation; and
- problems interfacing the functionalities on the cloud with others maintained in-house.

At an enterprise level, moving functionalities into the cloud is a concern relating to the obligation of customers' privacy and security to their personal data. One incident of data privacy breech at SaaS service model is Google Docs (Document and Spreadsheet product). In March 2009, Google disclosed that a breach had taken place in connection with its Documents and Spreadsheet products. Google sent a letter to certain users of these products stating that some of their documents, identified in the letter, may have been disclosed without their knowledge to persons with whom the users had shared documents previously, but whom the users had not authorized to receive the identified documents [48].

4.4.2 Cloud Portability

NIST defines cloud portability as the ability "of prospective cloud computing customers to move their data or applications across multiple cloud environments at low cost and minimal disruption" [56]. Termination of a cloud computing contract is a concern to enterprises as major cloud vendors such as Google and Microsoft Azure are using proprietary technologies and techniques that prevent the portability of cloud applications [57]. As such the ease of switching between cloud vendors is a disadvantage.

4.4.3 Laws and Regulation

Another concern to cloud computing is when a cloud customer undergoes investigation and litigation. Since data is no longer in-house, cloud vendors may have less motivation to object vigorously to attempts to obtain it, and may fail to notify (or delay notification) regarding those attempts [48]. Lastly, in the event of a provider bankruptcy, enterprise will need to ensure that the data stored on the cloud server will be recognized as owned by the user, and will not be viewed as assets of the provider that may be sold during the provider's bankruptcy proceedings and its privacy hereby compromised [48].

4.5 Summary

This section of the thesis provided literature reviews and comparisons of different articles relating to outsourcing. Cloud computing is a form of outsourcing as it shares the overall idea of using someone else's resource to perform IT services. Cloud-computing frameworks and models were also researched to outline different activities and costs associated with migration. Although the cloud computing model brings benefits, such as low initial cost, enterprises need a systematic framework to evaluate their existing in-house IT systems and determine which IT services will improve profit if they utilize cloud services. In the next section of this thesis, it outlines a framework for enterprises to evaluate their internal costs of operating their IT services and to determine whether using cloud IT services can increase profits.

Chapter 5 Proposed Cloud Migration Framework

This chapter introduces a novel framework for migrating the enterprise IT services to cloud computing environments. To aid an enterprise in making the migration decision, a series of guideline steps are first developed. Next, an optimization model is derived using a generic outsourcing profit maximization formula to reach the best migration decision in terms of which services to migrate and which to maintain at the enterprise's internal infrastructure. The model is then refined to incorporate coordination costs between vendors in addition to SLA penalties.

The remainder of this chapter is organized as follows. Section 5.1 explains the origin of how this framework is constructed. Section 5.2 proposes a novel framework that can be used to guide enterprises while defining their business strategies and objectives during the cloud migration process. The framework aids the enterprise in gradually decompose its business operations into business processes and map these processes to IT services.

Each step in the framework is explained further with suggestive guidelines. Section 5.3 proposes an optimization model for determining the mix of internal and external services. Sections 5.4 and 5.5 are dedicated to provide an example of a use case of this proposed framework, guidelines, and optimization formula, respectively, for an enterprise that provides IPTV services to its customers.

5.1 Comparison with the other literature frameworks

This proposed framework is inspired by literatures relating to outsourcing and cost optimization. Table 4 lists literatures and their comparison to this proposed framework.

Table 4: Comparison with other literature frameworks

Source	Comparison
Specifying services for ITIL service management [15]	In this proposed framework, it uses concepts from the Service Strategy and the Service Design phases of the ITIL Service Management model. The planning level of this proposed framework follows the "Service Strategy" phase of the ITIL model where it focuses on business objectives. The architectural level of this proposed framework extends the "Service Design" phase of the ITIL Service Management model by applying it to a cloud migration application. The ITIL's "Service Transition" and the "Service Operation" phase map to the proposed framework's Implementation level. In the implementation level of this proposed framework provides a guideline for enterprises to decommission their existing IT services while transitioning to cloud IT services. The ITIL's "Service Operation" phase maps to the "Monitor Cloud Services in compliance with defined SLA" step in the implementation level of this proposed framework where it observes service quality of IT

	services in a production environment. The ITIL's "Continual Service Improvement" refers to a cyclical application of this proposed framework where cloud migration can be applied in an iterative approach rather than a single pass through.
An e-business model ontology for modeling e-business [20]	In this proposed framework, it follows the three levels defined in the business logic triangle framework which are: planning, architectural, and implementation. This proposed framework extended the structure of the business logic triangle framework whereas the steps defined in the proposed framework are specific for the cloud-migration application.
First Steps to Creating a Cloud Computing Strategy for 2013 [53]	The Gartner's cloud-decision framework is structured as a flowchart where it has several decision points that branch of to different processes depending on whether it is a "yes" or "no" decision. The proposed framework has a different structure where it does not have any decision point and is intended for enterprises to use as a guideline. Also the Gartner's framework does not guide the enterprise that IT services can be migrated to the cloud in either SaaS, PaaS, or IaaS. The steps inside the Gartner's framework are similar to that of the proposed framework as follows. The Gartner's framework "Step 1: Prework" and "step 2: Business and Application Assessment" map to the "Define business objectives, goals, and strategies" process defined in the planning level of the proposed framework. The Gartner's framework "Step 3: Vendor Selection Process" and "Step 4: Mitigate Risk and Liability" map to the "Determine the scope of outsourcing by identifying which IT Services to migrate to clouds" process in the planning level of the proposed framework. Finally, the Gartner's framework "Step 5: Steady State" maps to the "monitor cloud services in compliance with the defined SLA" process in the proposed

	framework.
A quantitative model for understanding multiple vendor IT outsourcing [50]	The optimization model of the proposed framework extends the idea from [50] that to achieve the optimal profit, expense needs to be minimized. The optimization model extends [50] by taking into consideration of maximizing the revenue as well as minimizing the expense in the overall profit maximization model. Article [50] focus only on expense minimization where as the propose model calculate the optimal solution that will maximize the overall net profit. Literature in [50] introduced that there are overhead expenses when services are outsourced to multiple vendors. The optimization model proposed in this framework applies the coordination costs between two services i, j using matrix $\boldsymbol{Q_{ij}}$ when two IT services are served in two different clouds.
Solving the integrated product mix-outsourcing problem using the imperialist competitive algorithm [65]	Literature in [65] introduced a profit model for a general outsourcing strategy. The optimization model proposed in this framework extends [65] by expanding the "*OP*" operating expense variable to cloud computing application. The proposed framework expanded the *OP* variable as a function of to incorporate the cost of using the cloud service plus integrity, confidentiality, and availability loss that might occurred if a cloud vendor k had a security breach or service interruption.

5.2 Proposed Framework and Optimization Model

The objective of this proposed framework is to provide a guideline for enterprises through a series of steps while deciding which IT processes should be migrated to clouds and which ones should remain within the enterprise IT infrastructure. This proposed framework guides enterprises to start with a set of as-is analysis of the operation of its current IT processes, their current costs and constraints. These constraints are then transformed into a set of policies that guides the migration decision. This proposed framework consists of steps that are organized into three layers: planning, architecture and implementation. By following the steps starting from the top layer to the bottom layer of the proposed framework, it guides enterprises to decompose strategies into processes and from processes to IT services.

The purpose of the planning layer in this proposed framework is for enterprises to define cloud migration strategies and objectives at a conceptual level. The former include defining which part of the business can be migrated to clouds subject to security concerns, evaluating potential cloud vendors, and defining a transition plan for the enterprise to adopt cloud services. An "as-is" cost model is constructed based on the existing workflow and structure; this "as-is" cost model becomes the baseline for searching for SaaS, PaaS or IaaS solutions.

The purpose of the architecture layer in this proposed framework is for enterprises to search for cloud services that can satisfy business objectives. Enterprises use the existing workflow to find which processes can be candidates to migrate to clouds. Enterprises are suggested to define a data model for communication between cloud vendors and/or in-house applications. The data model along with the amount and frequency of how IT services interoperate are used to calculate the coordination costs of using cloud services based on inbound and outbound rates. It is at this level that the cloud-coordinate cost function is defined. Depending on the quality of the cloud services, additional expenses might be incurred due to different factor such as confidentiality, availability, and/or integrity losses; these are categorized as the total "expected loss". This step then drives the evaluation of risks associated with each vendor. It is also at this layer that the enterprise architects compare different cloud vendors, define SLA, and negotiate contracts. Once cloud vendors are determined, a "to-be" cloud-operation cost model is constructed to capture costs associated with available cloud migration scenarios.

Finally, this proposed cloud-migration-optimization model of the framework takes the calculated costs in the previous steps as inputs to determine the right mix of in-house and cloud services to be used while satisfying the net costs of the IT operations.

The implementation layer incorporates the cloud services into an enterprise's business processes according to the architectural design. At this layer, the enterprises' IT department achieves the interoperability of the processes using cloud resources. Integration of cloud services with in-house processes involves activities such as installation, code modification, migration of data and processes, training, and testing.

This proposed framework is depicted in

Figure 5.1, which sketches the necessary steps for deciding whether to migrate IT services based on costs. These steps will be discussed in details in the following sections.

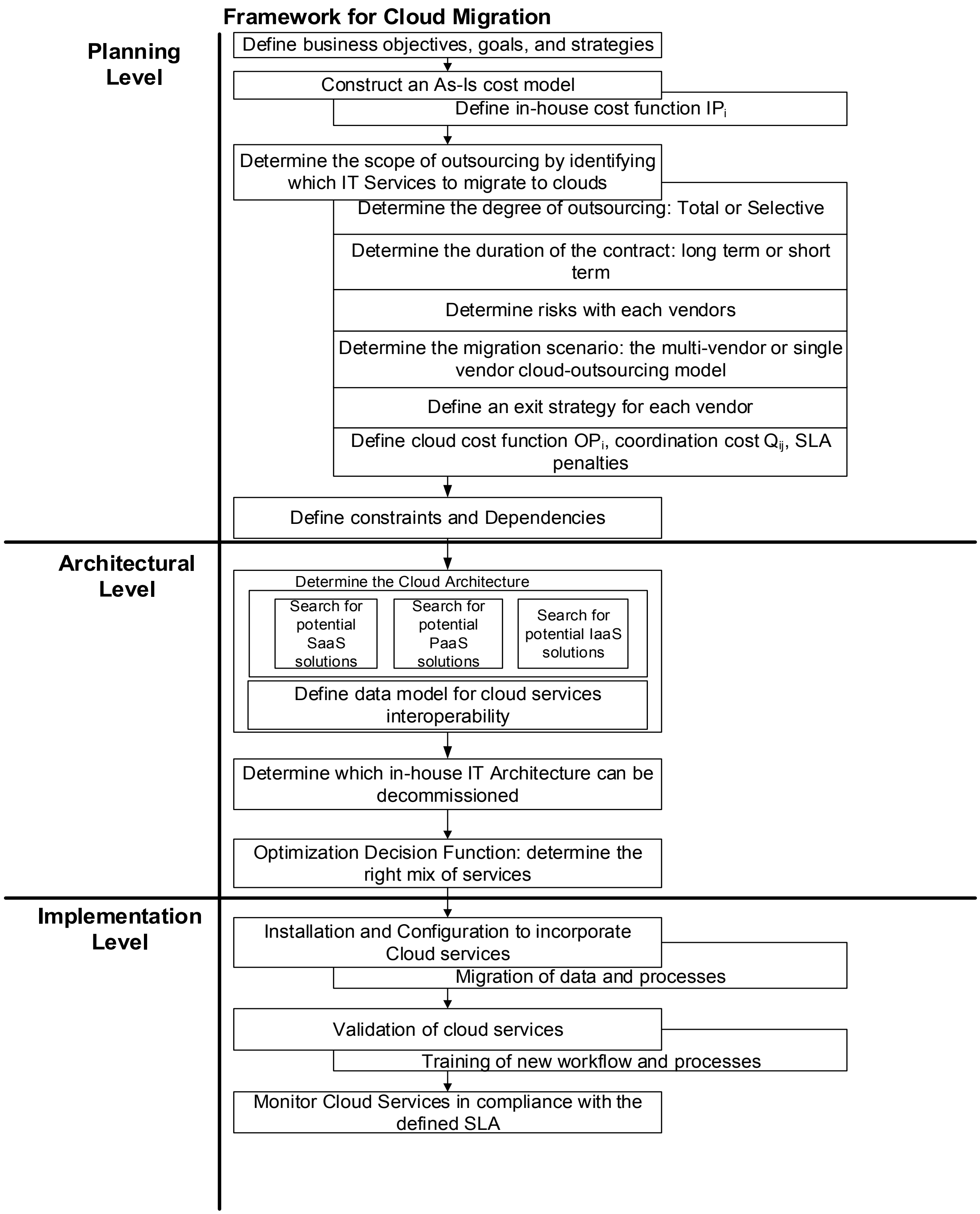


Figure 5.1: Proposed Cloud migration framework

5.2.1 Define Business Objectives, goals, and Strategies

This proposed decision framework for migrating IT services to the cloud targets enterprises that already have a pre-existing IT model. The first step in the framework is to define business objectives, goals, and strategies which are the foundation for managing the business and IT relationship. The purpose of this step is to set migration goals that will guide the rest of the steps in this proposed framework. These goals will also be used to validate the outcome of the migrated IT services.

At this stage, enterprises should start with the question "What do I gain by migration to clouds?" A brainstorming technique can be used to generate ideas amongst a group of IT executives. A mind map diagram is a visual outline of ideas which can benefit enterprises in documenting and organizing information from brainstorming sessions [40]. The goals that can be derived from this starting question are:

- Enterprises no longer need to worry about infrastructure
  - **Goal**: Reduce in-house IT infrastructure maintenance
- Enterprises no longer need to worry about security
  - **Goal:** Transfer the risk to cloud providers
- Enterprises no longer need to worry about hardware expansion to meet business demands
  - **Goal:** Leveraging cloud elasticity and fast up time of virtualized environment to manage hardware capacity
- Enterprises no longer need to worry about application development
  - **Goal**: Rely on SaaS cloud providers for already built software and focus on SaaS configuration to implement business rules
- Enterprises can focus more on core competency areas such as R&D
  - **Goal**: Develop new products to generate future revenue streams

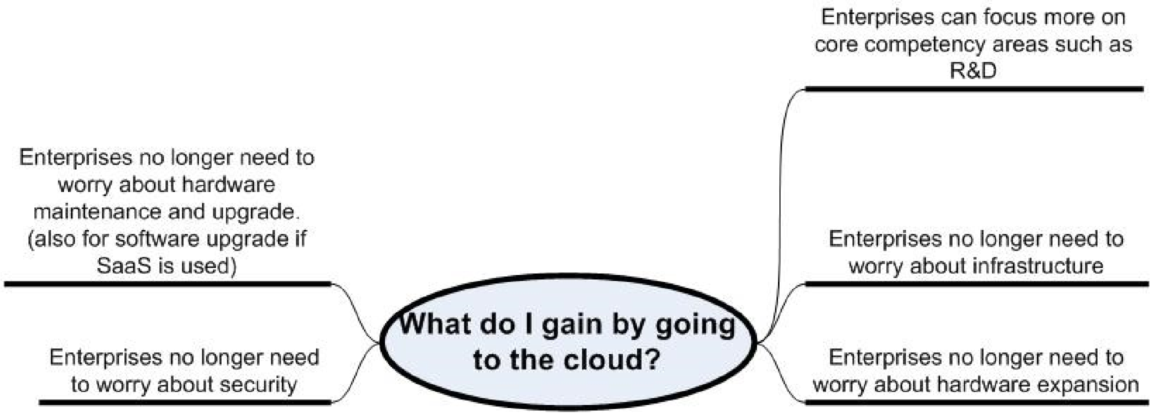


Figure 5.2: Mind map example to derive goals from outsourcing

The following table summarizes this strategy planning step:

Table 5: Define Business Objectives, goals, and Strategies – Inputs, techniques, and outputs

Inputs	Techniques	Outputs
- Existing IT Model	- Mind map - Brainstorming	- Goals for cloud migration

This step in the framework is the foundation for migration to clouds because these goals are what the cloud services need to satisfy.

5.2.2 Construct an As-Is Cost Model

In the previous step, the enterprise has defined the overall goals for migrating IT services to clouds. The enterprise then calculates the current as-is unit cost of each IT service in preparation for the next step which is to find more cost effective solutions in the cloud. To realize this step, the service-based cost model is employed which is defined and showed in [58] Figure 5.3.

The objective of this step is to determine the overall cost into a unit cost function $IP_i(d_i)$ for each service i based on demand d_i. First we identify the sources of costs which can be from external service, software, people, hardware, and facilities as shown in

Figure 5.3. Using the payroll system as an example of an IT service, it requires a banking service (external service), a payroll program (software), staff (people), servers (hardware), and office space (facilities). Secondly, we need to forecast the demand for each service.

Using the above steps, we construct the cost function $IP_i(d_i)$ that reflects the monetary cost associated with service i as a function of the demand d_i. This function depends on the recorded or forecasted quantity (or number of requests to a service). In general, more resources are needed to produce more units of a given service. As a result, the function $IP_i(d_i)$ is an increasing function in the demand d_i.

This function accounts for the enterprise's direct, indirect, and overhead costs associated with each service i. Direct costs are costs directly related and completely attributable to i, for example, the payroll software brought to run the payroll system is the direct cost associated with the payroll service. Indirect costs are shared across other services, for example, costs related to IT security are usually shared among all services within a single department. Finally, overhead are general administrative costs that cannot be directly attributed to a service, for example, the salary expense of executive staff. The indirect and overhead costs are divided amongst the forecasted demand of IT services plus a margin to derive a unit cost.

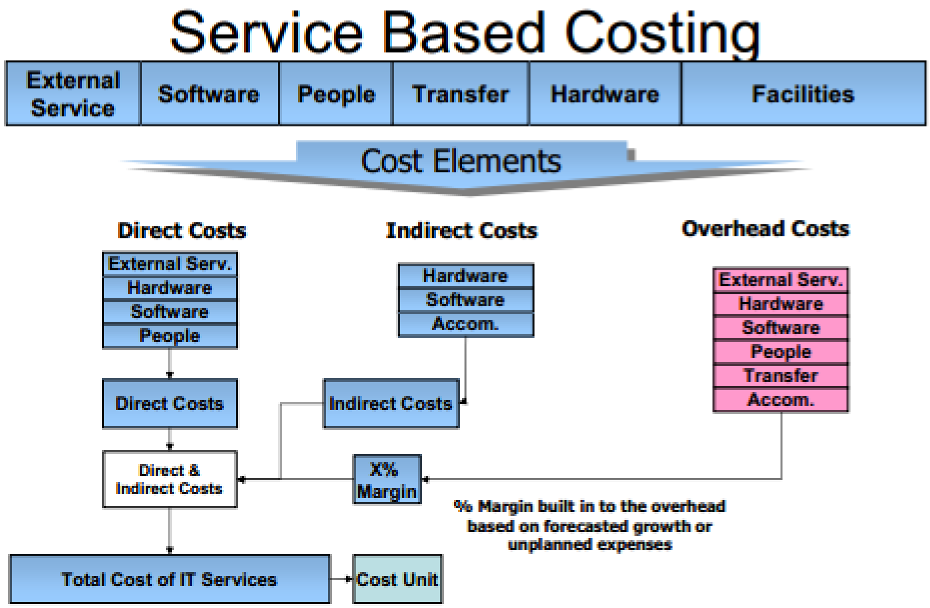


Figure 5.3: Service Based Costing [58]

A year of data from either forecasted or historical records is recommended to calculate this unit cost. Figure 5.3 shows a "% margin" which is used to account for unplanned expenses or deviation in forecast when forecast data is used. Later in this chapter, a proposed optimization engine for cloud migration is introduced and will utilize this cost function $IP_i(d_i)$.

The $IP_i(d_i)$ cost function varies depending on the service type. For example, if the cost of the service is fixed, the function is linear. Also, quadratic and staggering functions are used to represent more complex cost functions. A staggering function fits better for an IT service for taking the capacity limit into account; for example, if one web server can handle up to a hundred transactions per hour, then one hundred is the capacity limit of that web server. When there are more than a hundred transaction requests, an extra web server is required to handle the extra capacity. This adds a lump-sum value for the price of the new server when the demand exceeds the original capacity. As such, $IP_i(d_i)$ will be in an increasing-staggering function.

The following table summarizes this step:

Table 6: Construct an As-Is Cost Model – Inputs, Techniques, and Outputs

Inputs	Techniques	Outputs
- Goals for cloud migration - Forecasted demand - Overall costs	- Service based costing	- $IP_i(d_i)$ cost function of doing service i in-house according to demand d_i

5.2.3 Determine the Scope of Migration

The previous step of this proposed framework defined the migration goals and as-is cost. In this step, the business scope of cloud migration is defined.

5.2.3.1 Determine the degree of outsourcing: Total or Selective

The degree of outsourcing refers to whether an enterprise would want to keep partial operations within the company [59]. For example, when an enterprise works with the US government that contains NSA (National Security Agency) data, these applications and their related data may not migrate to the cloud. Total degree of outsourcing means that an enterprise migrates most or all of their IT operations to a cloud vendor. It is important to understand the existing IT governances and security policies to make a decision on which services to migrate to clouds.

Let d_i be the demand for service i, as defined before. We define two new variables x_i and y_i to represent the degree of outsourcing, where x_i is the number of service executions that should be migrated for service i out of d_i, while y_i is the number of service that should be operating within the enterprise. When y_i/d_i is less than 80%, it is regarded as selective migration. Whereas if y_i/d_i is greater than 80%, we consider this as a total migration scenario [60].

In the case of a selective migration, an additional communication cost might be incurred between two dependent services that are located in two different locations (i.e. one service in the cloud and the other within the enterprise IT). To capture this cost, we define the function $Q_{ij}(t)$ which represents the communication cost between service i and service j when service i is in a different location from service j.

5.2.3.2 Determine the duration of the contract: long term or short term

The duration of the cloud migration contract is an important factor for scope planning. When dealing with cloud migration, an exit strategy is required to get the data out of one cloud into another cloud or transfer the data back into an enterprise. There is an overhead cost associated with switching between cloud vendors such as transferring data out of the cloud, configuring the business rules in the new SaaS cloud, and testing the end-to-end solution. This overhead cost is categorized as "allocation" cost. As such, to minimize allocation cost, a long-term relation between an enterprise and cloud vendors is desired.

To capture the cost of operating in the cloud, we define the function $OP_i(x_i)$ which represents the outsourcing cost for x_i executions of service i. The allocation cost of i is encapsulated inside $OP_i(x_i)$. Once a service is in the scope of migration to clouds, an enterprise needs to define $OP_i(x_i)$ for that service i.

Later in this chapter, a proposed optimization engine is introduced and will utilize this external cost function $OP_i(x_i)$ to reach final migration decision.

<u>Cloud-vendor usage cost:</u>

There are different cloud-pricing models such as subscription based and pay-per-use based. When it is a subscription based pricing model, the $OP_i(x_i)$ cost function will need to divide the subscription price by the forecasted amount to generate a per unit price. In a pay-per-use pricing model, the $OP_i(x_i)$ cost function will incorporate this per unit cost.

The $OP_i(x_i)$ cost function is an aggregate of the vendor's cloud operating cost based on the pricing model, negotiation cost, adoption cost, and allocation cost.

Cloud Operation Cost:

As mentioned in a previous section, the pricing model for cloud can be subscription or pay-per-use base. For the subscription pricing model, the $OP_i(x_i)$ cost function needs to divide the overall subscription price per year by the annual forecasted demand to derive a per-unit cost. For the pay-per-use pricing model, the $OP_i(x_i)$ cost function can use this unit price without knowing the forecasted demand.

Negotiation Cost:

Enterprises are advised to compare different cloud vendors that can meet their requirements. The non-functional requirements of cloud services are suggested to be documented in a SLA that contains guaranteed levels of performance, uptime, and vendor responsiveness. Defining a precise SLA is important when dealing with external vendors. In some context around the SLA, enterprises are eligible for a service credit if a cloud provider misses the committed SLA [61]. Enterprises are suggested to monitor their cloud vendors SLA. One SLA monitoring framework is the Web Service Level Agreement (WSLA) which provides a standard to help enterprises and cloud vendors to specify SLAs in a flexible and individualized way, provides a system to provision resources based on service level objectives, provides a workload management system that prioritizes requests according to the associated SLAs, and provides a system to monitor compliance with the SLA [62]. The SLA is suggested to be formalized in a contract between the enterprise and the cloud vendor. The variable SLA_i is used in a proposed optimization function to capture the financial loss due to failure in cloud services. Enterprises are advised to have an architectural exit strategy in the event that they are not satisfied with a cloud vendor and that they need to transfer to another cloud vendor or bring the IT services back in-house.

Adoption Cost:

It is important to determine the risks associated with a selected list of cloud vendors. In an article by Bisong and Rahman [63], he recommends that due diligence should be conducted to determine the viability of the cloud service provider. Enterprises are encouraged to consider factors such as vendor reputation, transparency, references, financial (means and resources), and independent third-party assessments of vendor safeguards and processes [63]. The $OP_i(x_i)$ cost function can express this risk as a potential expected monetary loss due to failure in cloud resources. This monetary value is then divided by the forecasted demand of service *i* to derive a per-unit risk cost.

Allocation Costs:

At the implementation layer, the allocation costs consist of transforming the enterprise's internal process to use cloud services, refactoring software modules to exchange data with cloud vendors, training staff on the new processes, and testing whether these cloud solutions meet the business objectives. The $OP_i(x_i)$ cost function can express this allocation cost by estimating a monetary value to transform internal services to use cloud resources and divide this number by the forecasted demand of service *i* to derive a per-unit allocation cost.

In summary, the $OP_i(x_i)$ clouds-cost function is an aggregate per-unit cost of the vendor's pricing model, negotiation cost, adoption cost, and allocation cost. Similar to the $IP_i(y_i)$ in-house cost function, $OP_i(x_i)$ is also expected to be an increasing staggering function. The reason why it is a staggering function is that cloud resources have capacity limits and once that limit is reached, the next level is required to handle extra capacity. As such, both $OP_i(x_i)$ and $IP_i(y_i)$ are functions of quantity.

5.2.3.3 Determine the migration scenario: the multi-vendor or single vendor cloud-outsourcing models

During this analysis step, enterprises are advised to evaluate whether a multi-vendor or single-vendor outsourcing model is a better fit. Multi-vendor outsourcing means that the enterprise may incur extra costs for inter-cloud communication and coordination activities to manage this inter-cloud coordination [29]. We use the same function $Q_{ij}(t)$ to represent this additional cost for simplicity.

Finally, it is worth mentioning that some possible outcomes of this analysis outlined in this step are:

- Migrate entire IT systems and services to multiple cloud vendors
- Migrate partial IT systems such as the payroll business processes to the SaaS cloud (example ADP)
- Migrate partial IT infrastructure to the IaaS cloud
- Maintain application development in-house but migrate application deployment to PaaS cloud

Cost-benefit analysis is a technique used in this scoping step to determine if the cloud-migration cost meets the migration goals as previously defined. The outcomes of this step are cloud migration requirements, a list of IT services to migrate to clouds, cloud-resource-cost function, and communication-cost function.

The following table summarizes this scoping step:

Table 7: Determine the Scope of Migration – Inputs, techniques, and outputs

Inputs	Techniques	Outputs
- Goals for cloud migration - Governance and Security Policies - As-Is Cost - Forecasted demand for each service *i* - SLA for each service *i*	- Cost-Benefit analysis - Vendor comparison	- A list of cloud vendors that enterprises can migrate their IT services to - Which IT services are candidates for cloud migration - $OP_i(x_i)$ cost function of outsourcing service x_i - $Q_{ij}(t)$ communication cost function between service *i* and *j* - SLA_i penalty function if a service *i* cannot meet SLA - Cloud migration requirements

5.2.4 Define Constraints and Dependencies

In the previous step of this framework, we identified the users and workflows that meet the business objectives. In this step, we analyze the IT services required to support the IT processes that were defined in the previous step by defining constraints and dependencies.

The communication constraint for cloud architecture is that data packets have to be transmitted out of the LAN to the WAN before arriving at the cloud provider network; data transmission has to be fast enough to meet the SLA defined for these IT services. The SLA is a constraint for IT services that are tightly coupled that need to transmit data frequently in a short timeframe. To determine the relation between processes and their IT services is to determine how tightly coupled IT services are to each other.

By analyzing the mappings of processes to their IT services, dependencies are determined. The constraints to these IT services are that they must meet the defined SLAs. When these IT services are tightly coupled, to lower the risk of latency in inter-cloud communication, it is suggested that these IT services be migrated to the same cloud vendor or to do them all in-house.

The following table summarizes this scoping step:

Table 8: Define Constraints and Dependencies – Inputs, Techniques, and Outputs

Inputs	Techniques	Outputs
- As-is IT architecture - Workflows	- Analyze process-to-IT-services map	- A list of services that is best to be kept together. - SLA (constraints) and dependencies

5.2.5 Define the Cloud Architecture

The previous foundational steps of the proposed framework defined the conceptual business objectives, as-is cost, and cloud migration scope. The purpose of this step is to determine at which cloud deployment model a service can be migrated. The first step, at the architectural layer, is to identify whether SaaS, PaaS, or IaaS will be best suited for each service that is a candidate to migrate to the cloud. In the previous step, we identified constraints for some of the services to be kept together. Secondly, for each service that interoperates with other services, a data model needs to be defined. Thirdly, once the cloud services are defined, enterprises will need to update the stakeholder-interaction documentation since the use cases will be different.

5.2.5.1 Determine the Cloud Deployment Model

To determine whether to use SaaS, PaaS, or IaaS to implement the business needs is a challenge. An article published by Rackspace summarizes the recommendations of each cloud services [64]:

The prime candidates for an initial move to SaaS are:

- Generic offerings where the solution is largely undifferentiated. For example, an enterprise migrates its Microsoft Outlook and Microsoft Exchange server email service to Google GMail SaaS solution.
- Applications where there is a significant interplay between the organization and the outside world. For example, an enterprise migrate its email newsletter campaign software to use a customer-relationship-management (CRM) SaaS software such as Salesforce.com.
- Applications that have a significant need for web or mobile access. An example would be mobile sales management software.
- Software that is only to be used for a short term need. An example would be collaboration software for a specific project. For example, an enterprise migrates its JIRA project-collaboration-application with Confluence SaaS.
- Software where demand spikes significantly, for example tax or billing software used once a month. For example, an enterprise migrate its Intuit Quickbooks book-keeping desktop software with Quickbooks SaaS solution.

The following are scenarios where SaaS may not be appropriate:

- Real-time applications where fast processing of data is required; for example, virus scanning software that monitors web browser traffic.
- Applications where legislation or other regulation does not permit data being hosted externally; for example, applications that are constrained by NSA requirements.

- Applications where an existing on-premise solution fulfills all of the organization's needs.

The scenarios for choosing PaaS cloud service model are:

- PaaS is suggested for enterprises that have an existing data source – for example sales information from a customer relationship management tool, and want to create applications which leverage that data.
- PaaS is useful where developers wish to automate testing and deployment services. For example, a software-development enterprise uses PaaS to integrate their test scripts on their deployment environment.

The following are scenarios where PaaS may not be appropriate:

- Where the application needs to be highly portable in terms of where it is hosted.
- Where proprietary languages or approaches would impact on the development process. For example, an enterprise that develops firmware software on a specific chipset.
- Where a proprietary language would hinder later moves to another provider – concerns are raised about vendor lock-in. For example, a Microsoft .NET software development application platform can only be deployed to PaaS providers that support it.
- Where application performance requires customization of the underlying hardware and software. For example, firmware development in an automobile that is customized for that specific make and model.

The scenarios for choosing an IaaS cloud service model are:

- Where demand is very volatile – any time there are significant spikes and troughs in terms of demand on the infrastructure. For example, a retail-store website requires more servers to handle their demands during Christmas season.

- For new organizations without the capital to invest in hardware. For example, startup companies that do not have money to have their own data center may choose Amazon AWS to host their virtual machines.
- Where the organization is growing rapidly and scaling hardware would be problematic.
- Where there is pressure on the organization to limit capital expenditure and to move its expenses to operating expenditure.
- When infrastructure is only needed for a short period of time. For example, a company needs a super computer to analyze protein sequence for a trial project may choose to configure Google Cloud high-processing servers for a month.

The following are scenarios where IaaS may not be appropriate:

- Where regulatory compliance makes the offshoring or outsourcing of data storage and processing difficult. For example, applications under NSA policy have a constraint on their data to be stored in the USA.
- Where the highest levels of performance are required, and on-premise or dedicated hosted infrastructure has the capacity to meet the organization's needs.

It is suggested that enterprises follow the Software Development Lifecycle (SDLC) to properly design and test the cloud software by doing User Acceptance Testing (UAT). As such, the outputs of this step are acceptance criteria and test cases that will be used in the implementation stage to validate that cloud services met the requirements.

5.2.5.2 Define a data model for cloud services interoperability

Organizations at an enterprise level that follows the ITIL framework should already have their as-is processes and workflows documented. The purpose of this step is to derive an IT architecture to provide the same business goals with cloud services. The swim-lane and the UML sequence diagrams are suggested to document the interactions between processes in the workflow. Documenting process interactions involve defining the data model, which is defining the types of information being exchanged between the processes. A schema is suggested to document the information exchanged between processes. Since a process has inputs and outputs, it is important to use a schema to describe the inputs expected by the process and the outputs that the process will produce. Using the payroll system as an example, the following figure is a data schema represented in an UML class diagram.

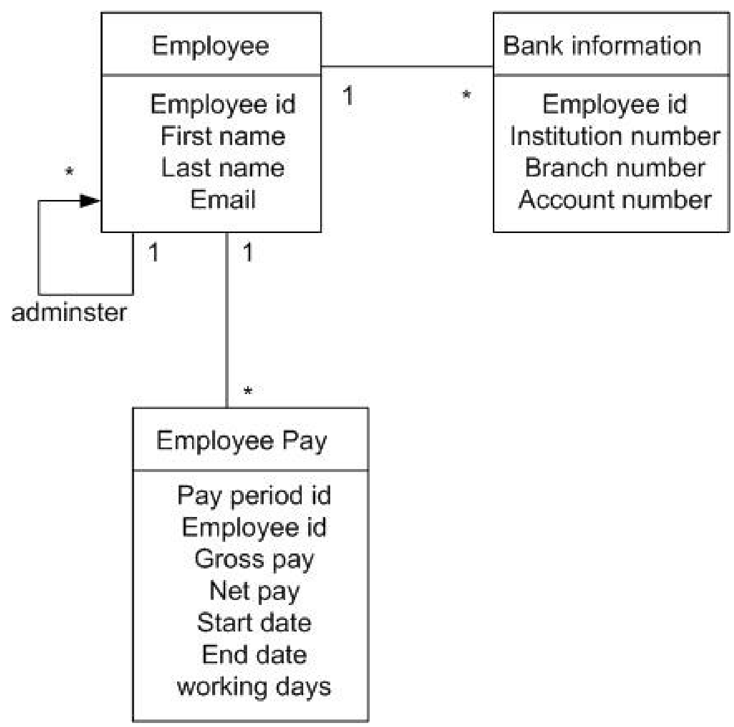


Figure 5.4: An example of payroll system Data Model

The business processes are organized as one or more supply chains of individuals, organizational units and companies working together in delivering products or services to the customers. The suggested tool for documenting stakeholders and their activities is to draw UML Use-case diagram. Using the payroll system as an example, the use case diagram is:

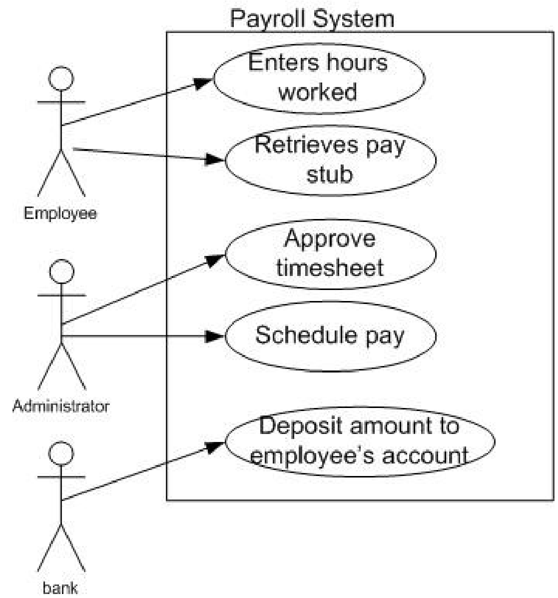


Figure 5.5: An example of payroll system Use Case diagram

In the previous steps the proposed framework covered documenting the migration goals, defining the scope of outsourcing, analyzing the as-is cost, and identifying IT services in a business workflow. In this step of the framework, we analyze at which level of the cloud service model that can meet the requirements with a lower cost. Continuing on the payroll example in previous steps, an enterprise can migrate their entire payroll system to a SaaS cloud; one SaaS cloud provider for payroll is ADP. An example of a use case for ADP payroll system for employee is:

Use Case: 1

Use Case name: Get payroll information

Actors: Employee

Description: An employee is provided with a login to the ADP payroll system where they can view and print past paystubs.

In this step, the data model defined in previous step is used to define the communication interface to exchange data between an enterprise and a cloud vendor. In the case of a multi-vendor cloud communication, the communication interface needs to define a protocol, a data model, and the frequency of this data exchange. For example, an enterprise may upload using a secured FTP port an updated list of employees' information once a day at 11:00 p.m. EST to the ADP system using an XML file that has the syntax validated against a predefined schema.

The interaction between an enterprise IT system and the cloud provider can be represented in a swim-lane diagram. Figure 5.6 elaborates on the payroll system where an enterprise uses a predefined data model when transferring employee and banking information to the Payroll cloud service; similarly, the cloud provider and banks have predefined data model to transfer the amount to the employees' bank accounts.

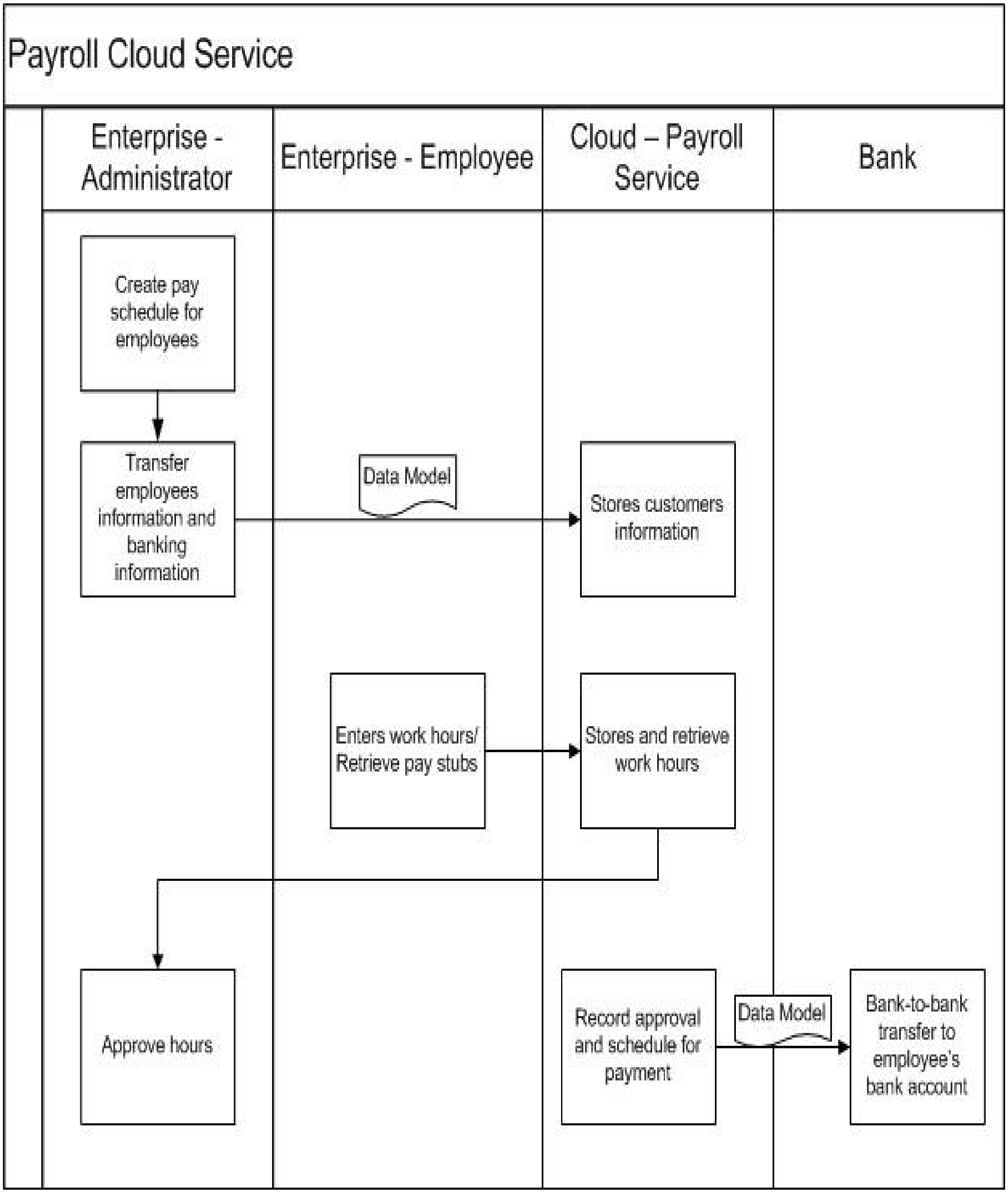


Figure 5.6: An example of a Swim-lane Diagram modelling the Payroll Cloud Service

The following table summarizes this architecture step:

Table 9: Define Cloud Architecture – Inputs, Techniques, and Outputs

Inputs	**Techniques**	**Outputs**
- Goals for cloud migration - Dependencies - List of IT Services to migrate - Data model - A list of services that is best to be kept together. - SLA (constraints) and dependencies	- Use case - Cost Benefit analysis - Class diagram UML diagram - Define acceptance criteria and validate cloud solution during UAT. - Define SLA	- Cloud vendor(s) and cloud solutions that will satisfy the business needs - Defined communication interface, protocols, and frequency to exchange data with cloud vendors. - Cloud solution IT architecture - A list of services that cannot be moved to the cloud

5.2.6 Determine which in-house IT Architecture can be decommissioned

One of the goals is to reduce IT infrastructure. The purpose of this sub step is to identify which of the in-house IT system can be obsoleted along with its associated IT infrastructure. This step can help in analyzing the cost savings for moving IT services to the cloud. In an ITIL model, a configuration management database (CMDB) is used to store a collection of IT assets (also known as configuration items) that are mapped to IT services. In this step of the process, the enterprise architect will analyze the as-is and to-be architecture to determine which IT assets can be decommissioned.

The following table summarizes this step:

Table 10: Determine which in-house IT Architecture can be decommissioned – Inputs, Techniques, and Outputs

Inputs	**Technique**	**Outputs**
- As-is IT architecture - To-be cloud solution IT architecture	- Analyze CMDB to find IT asset	- A list of IT services that can be decommissioned in-house. - A list of IT asset that the decommissioned IT services were using.

5.2.7 Optimization Decision Function: determine the right mix of services

In this step, the enterprise applies an optimization function given the inputs from previous steps to calculate the optimal mix of internal and external services that will maximize its profit. The optimization decision function uses the revenue function $P_i(d_i)$ for each service i and demand d, internal cost functions $IP_i(d_i)$, cloud cost functions $OP_i(d_i)$, coordination cost $\boldsymbol{Q_{ij}}$, and SLA penalty functions SLA_i to determine the optimal solution that will yield the greatest profit. It is worth noting that the optimal solution may not necessarily satisfy the entire demand d; depending on the penalty cost of not meeting the SLA, costs to serve additional demands, and revenue gain, it is more profitable to pay the SLA penalty than to pay for more cloud services if the additional revenues do not cover the expenses. The term "unmet demands" is used to describe the quantity of demand i that is not satisfied when the optimal profit point is reached. Section 5.3 is dedicated to describing the optimization model of this proposed framework in more details.

Table 11: Optimization Decision Function – Inputs, Techniques, and Outputs

Inputs	**Technique**	**Outputs**
- Revenue function $P_i(d_i)$ - Internal cost functions $IP_i(d_i)$ - cloud cost functions $OP_i(d_i)$ - coordination cost $\boldsymbol{Q_{ij}}$ - SLA penalty functions SLA_i	- Simulation - Optimization Decision Function	- An mix of internal and cloud IT services that will maximize the profit - unmet demands

5.2.8 Installation and Configuration to Incorporate Cloud Services

Once the architecture is designed, the next step is to incorporate the cloud services into the enterprise IT system. This includes configuring SaaS applications to meet business rules. The next step is to migrate data from in-house data center to cloud vendors. When migrating data, enterprises are suggested to use the Extract-Transform-Load (ETL) process. This ETL process is a guideline for data migration and cleanup to meet the required data model.

The following table summarizes this step:

Table 12: Installation and Configuration to Incorporate Cloud Services – Inputs, Techniques, and Outputs

Input	**Techniques**	**Output**
- IT transition architecture	- UAT - Training	- IT services are operated using cloud resources - Monitoring system to validate SLA of cloud vendors

5.2.9 Validation of Cloud Services

Once data and processes are defined, a small group of staff are asked to test and validate that the cloud solution meets business needs by performing UAT. Once UAT is completed, the transition to the new cloud processes include staff training and implementing a monitor system to validate that cloud services are in compliance with the defined SLA.

The following table summarizes this step:

Table 13: Validation of Cloud Services – Inputs, Techniques, and Outputs

Input	**Techniques**	**Output**
- IT transition architecture - Requirement document	- UAT - Training	- IT services are operated using cloud resources according to requirements

5.2.10 Monitor Cloud Services in Compliance with the Defined SLAs

Once the IT services are implemented and configured in the cloud, it is important to monitor these services that they are within the agreed SLAs. The monitoring module can be an internal or as a remote application. Internal monitoring applications are software running within an enterprise that measure the performance of its cloud IT services. Whereas remote-monitoring applications are software in the cloud that monitors its services; for some IaaS cloud vendors, monitoring and usage report of its services are provided to their customers. The metrics from these monitoring applications are used to determine when and how often cloud vendors failed to meet the SLA. When an IT service fails to meet its SLA, depending on the configuration of the monitoring tool, an alarm can be triggered to notify the IT department for actions.

The following table summarizes this step:

Table 14: Testing: Validation of Cloud Services with workflow – Inputs, Techniques, and Outputs

Input	**Techniques**	**Output**
- SLA	- Monitoring - Monitoring system requirements such as alarm definition and notification	- Alarms - Metrics and usage report - SLA penalty

5.3 Proposed Optimization Engine

The proposed optimization formula is derived from a generic business profit formula where profit equals to revenues minus expenses; the formula is then transformed to incorporate costs relating to cloud computing such as coordination costs between cloud vendors and service level agreement penalties.

5.3.1 Profit maximization with coordination cost

The generic business formula of profit is defined as the difference between revenue and expenses. The minimization formula outlined by Sounderpandian et al. [50] minimizes the expenses of outsourcing which consists of in-house operation expense, outsourcing operation expense, and coordination cost between services. This proposed optimization engine uses Sounderpandian et al. [50] expense-minimization technique together with revenue analysis to determine whether a service i is more profitable by migrating to the cloud. The formula from Sounderpandian et al. [50] is for generic outsourcing; to apply this to cloud migration, we need to elaborate expenses that are specific to cloud migration.

Nazari-Shirkouhi [65] introduces a variable *OE* (operation expense). This proposed formula expands OE to consist coordination cost for multi-vendor IT outsourcing from article by Sounderpandian et al. [50]. The combination of two formulas can be expressed as a maximization profit formula:

$$\text{Max}\left[\sum_{i=1}^{N}(d_i \times P_i(d_i)) - \left[(x_i \times OP_i(x_i)) + ((d_i - x_i) \times IP_i(d_i - x_i))\right] - \sum_{i=1}^{N}\sum_{j=1}^{N} Q_{ij} \times \left[x_i \times (d_j - x_j)\right]\right]$$

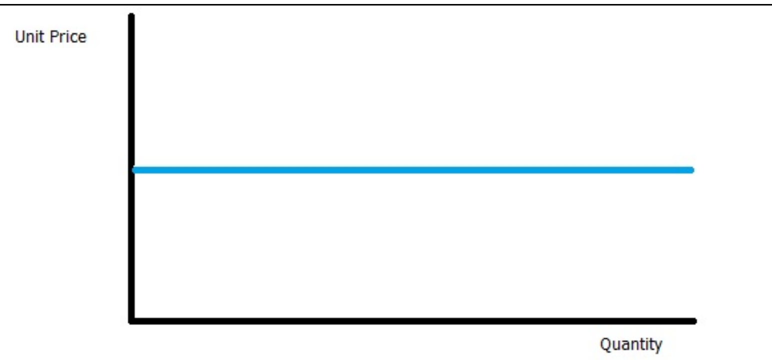
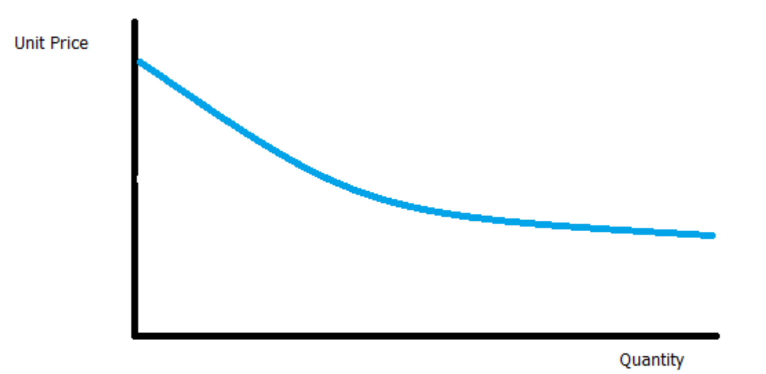
Where

- N be set of $|N|$ services.
- d_i is the expected demand for service i.
- $P_i(d_i)$ is the price that an enterprise can charge for d_i units of service i.
- x_i is the number of executions out of d_i to be migrated to the cloud where $0 < x_i < d_i$. x_i is an integer.
- $OP_i(x_i)$ is the outsourcing cost for x_i units of service i.

- $IP_i(y_i)$ is the in-house cost for y_i units of service *i*.
- $d_i - x_i$ is amount of in-house production for service *i*.
- Q_{ij} is the coordination cost between service *i* and *j*.

The reason that $P_i(d_i)$, $OP_i(x_i)$, and $IP_i(d_i\text{-}x_i)$ are functions of quantity is to model different types of pricing models. Figure 5.7 depicts three different pricing-revenue functions P_i based on demand.

A uniform-pricing function depicts a constant unit price that an enterprise charges to its customer regardless of the quantity being purchased. Secondly, the quantity-discount-pricing function is a decreasing function to encourage customer to buy more quantity; this decreasing function can be quadratic or a straight line. Thirdly, a variation of a quantity-discount-function can be in a form of a staggering function.

Uniform price (straight line) Revenue Function $P_i(d_i)$ $P_i(d_i)$=c, where c is a constant.	
Quantity-Discount-Revenue Function for $P_i(d_i)$ $P_i(d_i)=c\log(d_i)+k$, where *c*, *k* are constants.	

Quantity-Discount-Staggering-Revenue Function for $P_i(d_i)$

$P_i(d_i)=md_i+k$, for $0<d_i<a$

$P_i(d_i)=nd_i+k$, for $a<d_i<b$

$P_i(d_i)=od_i+k$, for $b<d_i<c$

m, n, o, k, a, b, c are constants.

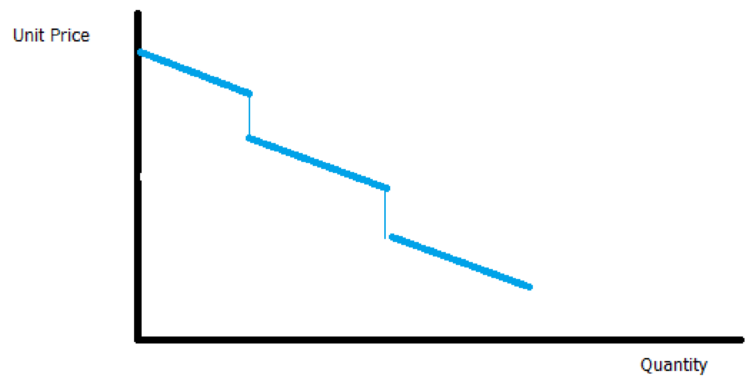


Figure 5.7: Types of Revenue function

Qij is a matrix where index [*i,j*] represents the cost for service *i* to communicate with service *j*. By definition, Q*ii*=0 for all *i* since there is no coordination costs associated if a service is communicating with itself; in matrix view, the diagonal indexes are zeroes. Note that this matrix is not necessarily symmetrical.

5.3.2 Profit maximization with Penalty Costs

Both Sounderpandian et al. [50] and Nazari-Shirkouhi [65] formulas did not take into account the cost associated with choosing one selection versus the other. A business definition of an opportunity cost is the loss of a potential gain when an alternative is chosen. In a scenario where there is limited resource capacity, a decision has to be made to avoid delivering a non-profit or low-profit service in order to leave enough capacity to serve another service for a higher profit. A Service Level Agreement (SLA) is a contractual promise that a service will be delivered in a certain condition; a penalty clause within a SLA is a cost to the vendor for failing to deliver the service [6]. This penalty cost is common to IPTV content delivery. To illustrate opportunity cost of two services, both services are assumed that they require the same amount of capacity:

Revenue for Service A = \$100; Cost of not meeting SLA for Service A=\$50

Revenue for Service B = \$101; Cost of not meeting SLA for Service B=\$0

Opportunity cost of selecting service A is \$101.

Opportunity cost of selecting service B is \$100+50=\$150.

In the previous section, we defined d_i as the demand for service i where x_i is the amount to use clouds' resources. To incorporate the use case where it is more profitable to pay the penalty cost and use those resources to deliver a higher revenue service, we introduce a new variable y_i as the amount of service i to do in-house which gives d_i-x_i-y_i as the amount of service i not delivered which are subjected to SLA penalty clause.

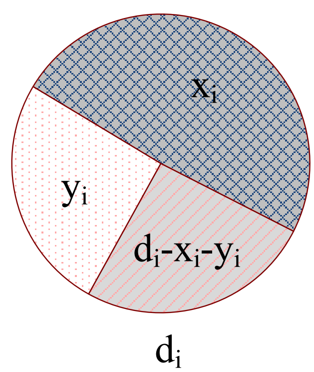


Figure 5.8: An example of unmet demands

The optimization can further be expanded to cover penalty costs through the new optimization function:

Max$\left[\sum_{i=1}^{N}(d_i - x_i - y_i)\times SLA_i\right]$, where SLA_i is the unit cost of not satisfying service i

To use this in our formula, we also need to define the total capacity availability and capacity required as resource constricts policy.

$\sum_{i=1}^{n} y_i * CAP_i \leq TotalInhouseCapacity$, where CAP_i is the resource capacity required for Service i.

We can now further elaborate the optimization formula to incorporate the penalty costs:

Max

$$\left[\sum_{i=1}^{N}((x_i + y_i) \times P_i(x_i + y_i)) - \left[(x_i \times OP_i(x_i)) + y_i \times IP_i(y_i)\right] - \sum_{i=1}^{N}\sum_{j=1}^{N} Q_{ij} \times (x_i \times y_j) - \sum_{i=1}^{N}(d_i - x_i - y_i) \times SLA_i\right]$$

5.3.3 Profit maximization with expected loss and multi-vendors

The Martens et al. [6] article defines outsourcing risks as expected losses which are confidentiality, availability, and integrity losses. In the previous section, the penalty cost of not delivering a service is considered as availability loss; when a service cannot be delivered to the customer because cloud vendor is unavailable, a penalty is paid to the customer according to the SLA. The variable SLA_i in the formula covers this availability loss. Confidentiality and integrity losses are potential outsourcing costs if a cloud vendor causes the enterprise to incur additional expense such as lawsuits over a security issue such as confidential data handling. Integrity loss can be in a form of data value corruption causing missing data. In the formula, we can decompose OP_i function to incorporate an estimate of confidentiality and integrity risks associated with this cloud vendor. In the previous section, the formula is based on a single cloud vendor. To incorporate confidentiality and integrity losses, we need to expand to multiple vendors to account for the scenario where a reputable cloud vendor has a lower risk.

In previous function, OP_i is a function of service x_i. To incorporate multi-vendor, OP_i is transformed as a function of both service x_i and vendor v_k.

Max

$$\left[\sum_{i=1}^{N}((x_i + y_i)\times P_i(x_i + y_i)) - \left[(x_i \times OP_i(x_i, v_k)) + y_i \times IP_i(y_i)\right] - \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} Q_{ij} \times (x_i \times y_j) - \sum_{i=1}^{N}(d_i - x_i - y_i)\times SLA_i\right]$$

Where $OP_i(x_i,v_k)$ = $OP_i(x_i)$ + IntegrityLossEstimate$_{vk}$ + ConfidentialityLossEstimate$_{vk}$ + AvailabilityLossEstimate$_{vk}$,
v_k is cloud vendor k for service x_i where $1<k<K$.

IntegrityLossEstimate$_{vk}$ is a monetary value of the cost of customers loss due to lack in confidence, regulatory penalty costs, and the estimated labor costs for restoring information integrity loss from vendor v_k [66]. The integrity loss calculation is the sum of the annual revenue loss from service i due to vendor v_k mistakes, a regulatory penalty cost of $1 million dollar (as an example), and the labour costs of staff to repair a major information integrity incident. The annual revenue loss from service i is derived by the annual revenue from each customer of service i multiple by the percentage of customer loss. This annual-integrity-loss value is then divided by the estimate demand of service i per year to derive the IntegrityLossEstimate$_{vk}$ per-unit loss.

ConfidentialityLossEstimate$_{vk}$ is a monetary value of the confidentiality loss from a vendor v_k. The calculation of ConfidentialityLossEstimate$_{vk}$ is the annual revenue loss from customer defection due to loss of information confidentiality and dividing it by the estimated annual demand for sevice i to derive a per unit cost.

AvailabilityLossEstimate$_{vk}$ is a monetary value of service interruption if a cloud vendor v_k failed to deliver the service causing the enterprise to incur losses such as damages to a company's reputation and customers closing their accounts due dissatisfaction.

5.4 An example: IPTV Case Study

The IPTV service can enhance the viewer's experience by tailoring contents based on a viewer's region. The IPTV service offers the flexibility for viewers to request videos on demand (VOD); some contents are dynamic which means that data are time sensitive. News feed, weather, and stock quotes are considered dynamic data. This data is stored on the Content Delivery Network (CDN) which contains content servers that are geographically spread [67]. The IPTV service offers a “triple play” service which includes video (broadcast TV, Video on Demand (VoD)), voice (Voice over IP), and data. The component that consumes the highest bandwidth is VoD. One strategy to offset the load is through replication to a nearby content server that is geographically closer to the viewers to reduce network latency [68].

IPTV service quality is important to the viewer; as such, delivering the content under the defined Service Level Agreement (SLA) is important while optimizing profit to reduce network latency by storing contents closer to the viewer. To reduce network latency in CDN, contents are replicated in many surrogate servers that are geographically located around the world. Figure 5.9 illustrates that the origin server contains the master data where it is replicated in the CDN with multiple surrogate servers around the world where localized clients connect to it [69]. A redirector service inside the CDN determines which surrogate server to deliver the content to the end users.

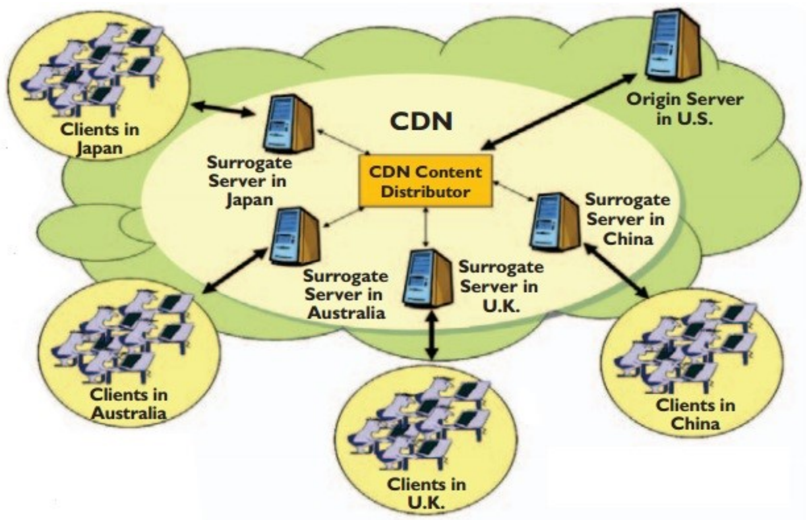


Figure 5.9: An example of Surrogate Servers in CDN [69]

In a cloud environment, surrogate servers can be virtualized and brought up on-demand to handle loads on the system; the Virtual CDN manager is responsible for detecting significant changes in network conditions and triggering the necessary CDN resources adaptation [7]. As such, the CDN manager is an IT service of the overall CDN architecture. Figure 5.10 depicts the relationship between the redirector and the virtualized surrogate servers. When a request R_i for a digital content is received by the CDN, the redirector determines in its database which surrogate servers SS_i contain these data and the cost $C(R_i, SS_i)$ associated for using the network link between the end user and server SS_i. If there are no surrogate server that contains the requested content R_i, or if the cost to use the link $C(R_i, SS_i)$ is greater than that of $C(R_i, OS)$ while meeting the SLA, the origin server OS is used.

R_1 represents the request from client.

C is the cost function associated with the costs of the link in the graph

OS is the Origin Server (which host the content)

SS_i represents the surrogate server that contains the content R_1. Note that a surrogate server SS_i may be dormant at time t until the virtual CDN manager brings it up.

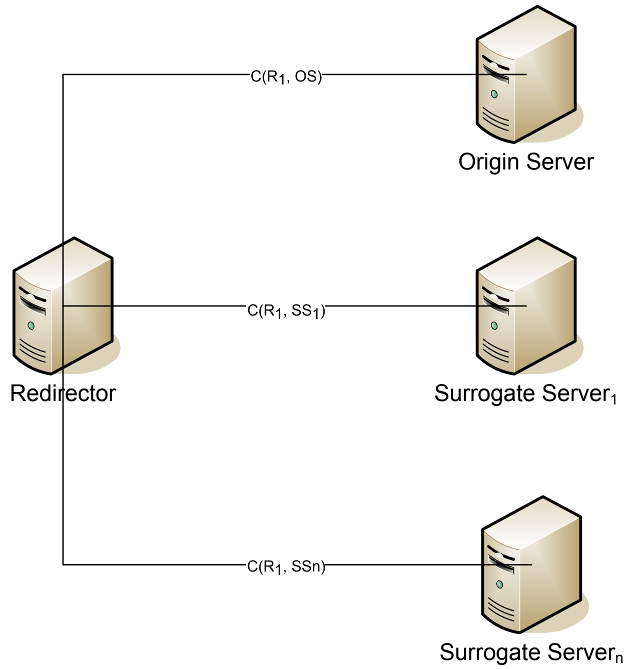


Figure 5.10: Graph representation of surrogate servers cost in decision making for content manager

5.4.1 Applying the Proposed Framework to the IPTV Service Scenario

This case study describes a decision model for IPTV service provider to maximize profits for outsourcing the Content Delivery Network (CDN) service to the cloud. Cloud CDNs is an approach that uses cloud resources (cloud storage) to reduce costs associated with implementing content-delivery services [70]. To apply this IPTV case study to this proposed framework, an IPTV service provider defines its cloud-migration objective to leverage cloud computing for its content delivery service, as it is too costly for IPTV service provider to have data centers in different countries. A benefit of using the cloud CDN is the flexibility of to add new surrogate servers based on demand and server capacity by programmatically to create, remove or change the location of the CDN components. Also, it enables an isolation of the virtual structure, allowing multiple instances of CDNs in a multi-provider scenario [7]. But, most importantly, the surrogates can be repositioned, adapting the infrastructure to new user demands and resource usage states. Thus, during a flash crowd event, for example an Olympic game, it should be possible to reposition surrogate servers closer to the source of the event to decrease the startup delay experienced by clients. The Virtual CDN Manager is used for management level decisions and changes. The Virtual CDN Manager receives information from the CDN Resource Monitor including important network traffic metrics, and based on its policies, may therefore decide whether and how to perform adaptation such as the startup and teardown of surrogate VM servers. Note that there are some VM servers in the CDN that are already started (hot standby) to decrease VM startup delay [7].

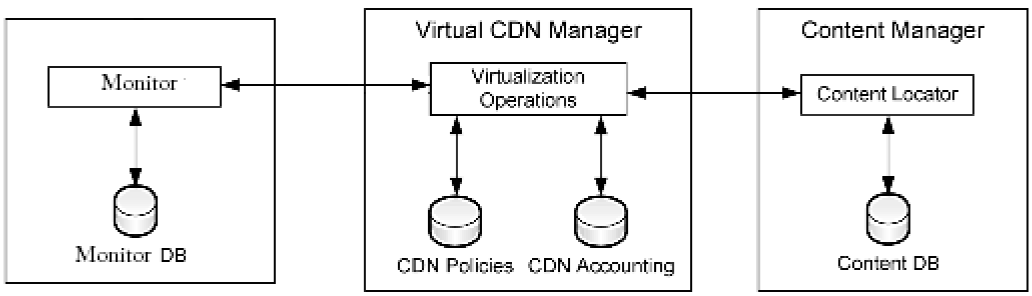


Figure 5.11: Virtual CDN [7]

The scope definition step for the surrogate-server component consists of account, network monitoring, redirector, and content delivery services. In this proposed framework, these set of services form the set *N*. For each of these services in set *N*, the IPTV provider defines $IP_i(d_i)$ functions which is the in-house costs that consist of facility (data centers), hardware (servers), human resources (staff to maintain the servers and software), and software costs. As suggested in this proposed framework, an IPTV provider is encouraged to use the past year of record to determine $IP_i(d_i)$. In summary, the setup for using this proposed framework is:

Table 15: IT Services of an IPTV Provider in a CDN

Services	Functions	Description
1. Accounting	$IP_1(d_1)$	The internal cost of managing the billing service for activities in the CDN. This involves the revenue to collect from end users and the expense to pay out for using the CDN for demand d_1.
2. Network monitoring	$IP_2(d_2)$	The internal cost of the Virtual CDN Manager service to monitor the network for demand d_2. Virtual CDN Manager is a piece of software to dynamically scale up or down vitalized surrogate servers.
3. Redirector	$IP_3(d_3)$	The internal cost of the redirector service to maintain a database for registering the contents in each surrogate servers so that the correct server is select to create a link to the end user for demand d_3.
4. Content delivery	$IP_4(d_4)$	The internal cost of using surrogate server to deliver service to the end user using the network link for demand d_3.

5.4.2 Expenses in a CDN

The CDN consists of multiple components to achieve the objective of delivering content to the viewer according to SLA. Data in a CDN is replicated in content servers based on their access frequency (also known as Hit Rate) [69]. CDN providers charge their customers (owners of the content) according to the traffic delivered by their surrogate servers to the clients. Factors that affect the pricing of CDN services are:

- Bandwidth cost;
- Variation in traffic distribution;
- Size of content replicated over surrogate servers;
- Number of surrogate servers;

- Reliability and stability of the whole system; and
- Security issues of outsourcing content delivery [69].

In this example, the IPTV provider uses a multi-vendor IaaS solution to implement its surrogate servers in different countries. Akamai is an example of a cloud vendor that has data centers in different locations that can serve as a redirector into their cloud CDN [71]. The cost to use the cloud servers are expressed as $OP_i(x_i)$. For example, $OP_4(x_4, v_k)$ is the network-delivery cost function of using the link in a cloud vendor v_k: $C(R_1, SS_n)$ as depicted in Figure 5.10. Note that a surrogate server can be a virtual machine that is on stand-by; in Amazon Web Services, the term “reserved instances” refers to virtual machines that are already configured that can be turned on when threshold reaches a certain level. As such, in this scenario, $OP_4(x_i,v_k)$ function is a staggering function where once the limit of a surrogate server capacity is reached, there needs to be an added cost to bring up the new virtual machine instance to service the next m requests where m<n. The decision function will need to take into consideration that $OP_4(x_i,v_k)$ is a non-uniform and non-linear function; to get the extra revenue p_i, we need to spend more operating expense. If we are just considering revenue and expense, it means that the revenue of subsequent request x_i should be greater than the extra $OP_4(x_i,v_k)$ to bring up the reserved instance. The $OP_4(x_i,v_k)$ is a complex function that needs to factor in the content migration, distribution, and content update costs. Content migration cost refers to the cost to transfer the media into the cloud CDN. Distribution cost is the cost of using the cloud-surrogate server to push the data to the end users. Content update cost depends on the type of data. Static content means that it does not change overtime; for example, a movie file, whereas news, maps or social networks information are dynamic content. The dynamic third party content changes periodically and once locally stored does not represent real-time service; hence, anytime a dynamic third party content is requested IPTV provider make sure to deliver the real-time version that complies with SLAs [72].

The CDN profit-maximization-decision model of which path surrogate server to redirect the request can be represented as a graph where nodes represent origin or surrogate content servers and lines between nodes represent a cost function for this link

[73]. Then, the profit-maximization-decision model is to calculate which path to take. The redirector will decide which surrogate server SS_i to redirect the request to serve the client request R_i (as denoted in Figure 5.10) based on the information from network monitoring service, distance to the client of the request, and cost of using the surrogate server SS_i.

When content is migrated to a cloud CDN, the confidentiality loss is that this content may be pirated or that the cloud provider network is vulnerable to hackers and may be accessed illegally. For availability loss, when video streams are not delivered to the customer, IPTV provider will incur the SLA penalty. However, IPTV provider can also establish SLA with the cloud provider. In this step, the SLA function is developed to determine the potential loss of not meeting a request that will be used by the overall decision function. Integrity loss for IPTV providers is that the content stored in the CDN is not retrievable. The IPTV provider will also need to define the functions $SLA_{inbound}$ and $SLA_{outbound}$. The $SLA_{inbound}$ refers to the penalty cost of not delivering the content within the SLA. The $SLA_{outbound}$ refers to the revenue from select service/network providers for not being able to deliver the content within the SLA. We can define an overall function SLA_i as $SLA_{inbound}$ - $SLA_{outbound}$.

The replication process can happen between two surrogate servers rather than from the origin server. When there is outbound traffic between two different cloud vendors' network, there is a coordination cost. It is worth mentioning also that a video content contains both audio and video streams; there can be two different surrogate servers representing service x_i as audio stream and service x_j as video stream. Q_{ij} is the coordination cost between two different surrogate servers in the cloud. The Q_{ij} communication cost is determined by extracting the unit price from the cloud provider pricing model.

In summary, the external cost function for using this proposed framework is:

Table 16: IPTV Service Provider External Costs

Services	**Expense Functions**	**Description**
1. Accounting	$OP_1(x_1)$	The external cost of managing the billing service for activities in the CDN.
2. network monitoring	$OP_2(x_2)$	The external cost of the Virtual CDN Manager service to monitor the network in the cloud environment. A PaaS example is Amazon CloudFront that enables content providers to provision capacity from Amazon's cloud resources in a pay as you go manner [74].
3. Redirector	$OP_3(x_3)$	The external cost of the redirector service to maintain a database mapping of its contents and surrogate servers.
4. Content delivery	$OP_4(x_4, v_k)$	The external cost of using surrogate server to delivery service to the end user using the network link for demand x_4 in cloud vendor k.
	SLA_i	The difference between the penalty costs of not meeting the SLA defined by service i and the possible revenue from cloud provider for not delivering the SLA.
	Q_{ij}	The coordination cost between service i and service j.

5.5 Profit Maximization for IPTV CDN Decision

The IPTV CDN surrogate-server selection of whether to use its origin server or an outsourced surrogate server can be deduced from a generic optimization model which is to maximize profit. The goal is to determine the quantity of requests that is profitable to use the in-house origin server to perform service i and the quantity needed to use the surrogate servers in the cloud using function $OP_i(x_i, v_k)$.

From the previous section, we deduced this mathematical representation:

Max

$$\left[\sum_{i=1}^{N}((x_i + y_i) \times P_i(x_i + y_i)) - \left[(x_i \times OP_i(x_i, v_k)) + y_i \times IP_i(y_i)\right] - \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} Q_{ij} \times (x_i \times y_j) - \sum_{i=1}^{N}(d_i - x_i - y_i) \times SLA_i\right]$$

Where

- N is the number of CDN services. They are billing, network monitoring, redirector, and content delivery. In this example, we will consider that a movie contains multiple segments.
- d_i is the expected demand for service i.
- $P_i(d_i)$ is the revenue from serving IPTV content.
- x_i is the amount that is handled by cloud-surrogate server
- y_i is the amount of IPTV content that IPTV provider is transmitting the information using its origin server (subjected that origin server has capacity and can meet the SLA).
- $OP_i(x_i, v_k)$ is the outsourcing cost of using the surrogate server in cloud vendor k.
- $IP_i(y_i)$ is the cost of delivering the IPTV content using the origin server.
- Q_{ij} is the coordination cost between two cloud vendors for replicating content in another cloud.
- SLA_i is the different between the penalty costs of not meeting the SLA defined by service i and the possible revenue from cloud provider for not delivering the SLA.

It is worth mentioning that "$d_i - x_i - y_i$" is the unmet demand of service i that we are not fulfilling. In the event that there is a capacity limit, the optimization model will determine which IPTV requests will generate more revenue with respect to paying penalty costs for requests that will generate less. This penalty represented in the formula SLA_i.

5.6 Summary

In this chapter, we have explained the steps in the proposed framework for enterprises to migrate their IT services to cloud services and how to adapt it to maximize profit. The proposed framework begins with guiding enterprises to define goals and scope for cloud migration at the planning level.

During the planning level of the proposed framework, enterprises define the unit cost function $IP_i(d_i)$ of doing these d_i IT services in-house. Similarly, an external cost function $OP_i(x_i)$ is defined to do these x_i IT services in the cloud. Besides the usage cost of the cloud service, we also need to account for communication and SLA non-compliance cost. The communication cost function $Q_{ij}(t)$ is defined to determine the added cost of transferring data between different cloud vendors or between cloud vendor and the enterprise network. The SLA_i cost function accounts for costs associated with penalties not delivering a service. The profit maximization formula uses all the cost functions to determine the optimal mix of in-house and cloud services of service i.

During the architectural level of the proposed framework, the enterprise searches for different cloud service model that can satisfy the business objectives. Data models are also defined in this step to set a standard for data communication between cloud services. From this architectural analysis, a list of potential cloud vendors that can satisfy the requirements is defined. Once the to-be architectural design is determined, a list of in-house IT services to decommission is determined.

At the implementation level of the proposed framework, the enterprise integrates their processes using the cloud services. This framework suggests that enterprises implement their monitoring services for breach in SLA.

Finally, this chapter illustrated the use of this proposed framework using an IPTV as a case study. In an IPTV infrastructure, it contains a content delivery network (CDN) to deliver the media to the viewer. The CDN network can use the proposed profit-optimization formula to determine whether to utilize a surrogate server in the IaaS cloud network or to use its internal server subject to the SLA requirement to deliver the content within a given time.

Chapter 6 Performance Evaluation

This thesis uses an empirical method by designing an experiment to solve the decision problem to optimize the use of virtualized servers on external cloud with IPTV content server as a case study. Matlab is used in this experiment to simulate the N services; coordination costs are simulated which represents costs associated when service *A* is hosted in a different cloud vendor than service *B*. This experiment is based on an "activity based view" where IT services support business processes [6].

The simulation tests that the optimization model correctly calculate the optimal set of x_i and y_i that will yield the maximum profit using different $OP_i(x_i)$ cost functions. The variables x_i and y_i represent the quantity of service *i* to outsource and quantity to perform in-house respectively. The three different $OP_i(x_i)$ cost-function patterns that are used in this simulation are constant functions, linear functions, and stagger discontinuous functions.

The assumption for this simulation is that all services take one unit of time to complete the operation. The objective of the simulation is to show that the formula is able to determine the correct combination of number of units to use clouds' resources for service *i* that will maximize net profits. If services take more than one unit of time, it becomes a scheduler optimization problem of which there is numerous research already done using algorithms such as Long Processing time (LPT), and Wilkerson-Irwin [75]. However, this scheduler optimization problem is not in scope for this simulation.

6.1 Validation of Simulation

To validate the optimization model, a Matlab program is written to generate simulation results. The Matlab program uses the "fmincon" function to find the local maximum of the proposed model. Each scenario has a table that summarizes the net profits, revenues, and expenses for three cases. The simulation validates that the proposed model will return the optimal set of x_i and y_i in these scenarios:

- Scenario 1 simulates a subscription pricing model by using constant revenue and cost functions.
- Scenario 2 simulates a pay-per-use pricing model by using linear cost functions.
- Scenario 3 simulates a stagger pricing model to illustrate that once a capacity limit is reached, additional servers are required to serve unmet demands. This scenario models the IPTV content server situation that additional VM is required to fulfill extra requests on the system.

Three cases are analyzed and graphed for each scenario:

- Case 1: Outsource all services
- Case 2: Perform all services in-house
- Case 3: Using the proposed model

For all three scenarios, the following parameters are used:

Number of Services

n=3

Three services will be used to validate the Matlab.

Demand

d_1=100

d_2=100

d_3=100

The demand for all three services is one hundred units.

Coordination cost matrix

Q_{ij} =

From/To	Service 1	Service 2	Service 3
Service 1	\$0	\$9	\$3
Service 2	\$9	\$0	\$5
Service 3	\$1	\$1	\$0

6.2 Scenario 1: Subscription Pricing Model Simulation

The purpose of this scenario is to show that the optimization formula is able to find the revenue maximization point by simulating a fixed-cost pricing model. In a fixed-cost pricing model for cloud computing, the subscription fee is constant. Since the proposed model requires a per-unit cost, this value is calculated by dividing the subscription price by the forecasted demands. In this scenario we use \$90 for external per-unit cost and \$80 for internal per-unit cost.

The equations representing constant revenue function $P(d)$, constant external cost function $OP(x)$, and constant internal cost function $IP(y)$ used for this scenario are

$P(d) = \$500, \text{ for all demand } d$

$OP(x) = \$90,\ \textit{for all units of } x$

$IP(y) = \$80,\ \textit{for units of } y \leq 50$

(Where 50 units is the capacity limit for internal resources)

In this scenario the results are:

Table 17: Simulation results for scenario 1: Subscription Pricing Model Simulation

Description	**Net Profit**	**Revenue**	**Net Expense**	**Coordination Costs**	**Unmet Demand**
Outsource all services	\$123,000	\$150,000	\$27,000	\$0	d_1-x_1-y_1=0 d_2-x_2-y_2=0 d_3-x_3-y_3=0
Perform all services in-house w.r.t. capacity limit	\$63,000	\$75,000	\$12,000	\$0	d_1-x_1-y_1=50 d_2-x_2-y_2=50 d_3-x_3-y_3=50
The proposed model	\$123,400	\$150,000	\$26,600	\$1,100	d_1-x_1-y_1=0 d_2-x_2-y_2=0 d_3-x_3-y_3=0

The following graphs were generated from a Matlab simulation. The independent variable is d_1 and the dependent variables are the profit, revenue, and expense. The other variables d_2, d_3, x_2, x_3, y_2, and y_3 are kept constant.

From the simulation in Matlab, the optimal set of x_i (number units to utilize cloud services) and y_i (number of units to perform in-house), for this scenario are

$x_1 = 50$

$y_1 = 50$

$x_2 = 50$

$y_2 = 50$

$x_3 = 50$

$y_3 = 50$.

In this scenario, $IP(y_i) < OP(x_i)$, for $x_i \leq 50$, the cost of performing services in-house is cheaper than using cloud services; as such, the optimization function maximize the usage of internal resources before using cloud resources. This 50 unit capacity limit for the in-house resources is illustrated in the following graphs. The graphs illustrated in the next sections show the findings when the demand d_1 varies from 0 to 100.

6.2.1 Demand Vs. Revenue

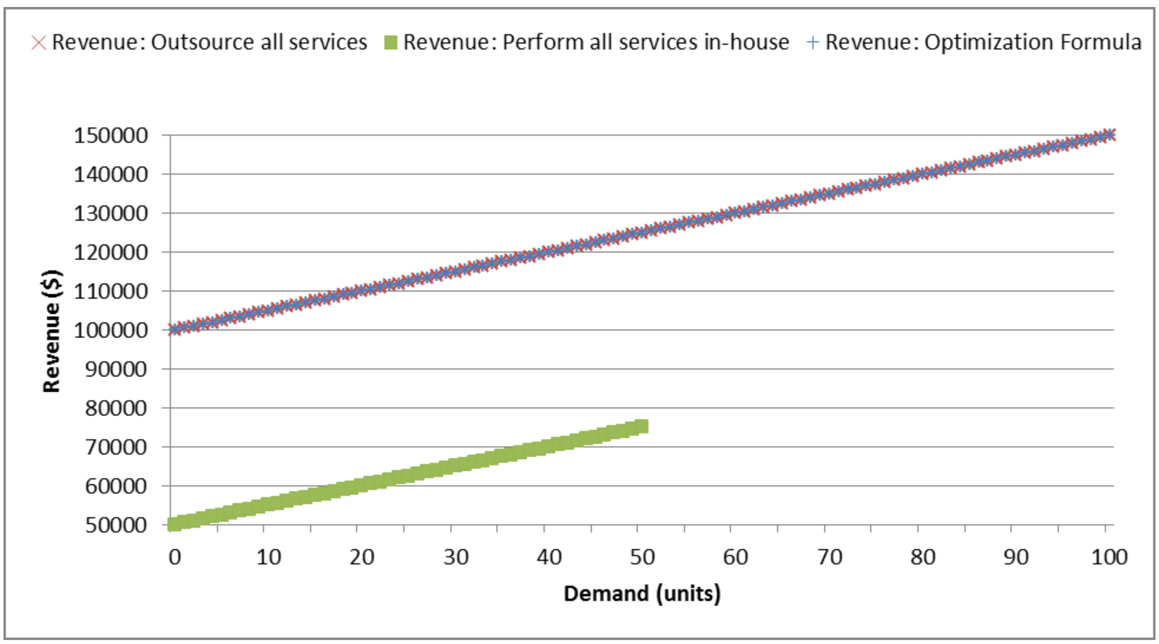


Figure 6.1: Demand vs. Revenue graph for Scenario 1: Subscription Pricing Model Simulation

Figure 6.1 depicts a demand vs. revenue graph that plots the results for these three cases: revenue where all services are outsourced, revenue where all services are performed in-house, and revenue using the proposed model.

The revenue results for both "outsource all services" and "optimization formula" cases, depicted by x and + in the graph respectively, have baseline revenue values of \$100,000. Since this simulation is only varying values for demand d_1, this baseline-revenue value is derived from values d_1=0, $d_2=100$ and $d_3=100$, which equates to $P(d_2) + P(d_3)$ = \$100,000.

The revenue results for "perform all services in-house", depicted by ■ in the graph, have a baseline revenue value of \$50,000. This baseline-revenue value is derived from values y_1=0, $y_2=50$ and $y_3=50$, which equates to $P(y_2) + P(y_3)$ = \$50,000. The resource capacity limit is 50 units for this "perform all services in-house" case; as such, there are unmet demands of 50 units for each service i. Unmet demand is d_i-x_i-y_i=50 for $1 \leq i \leq 3$.

The reason that the revenue results for all three cases are increasing in this scenario is from the direct correlation between the revenue and quantity of x_i service.

6.2.2 Demand Vs. Expense

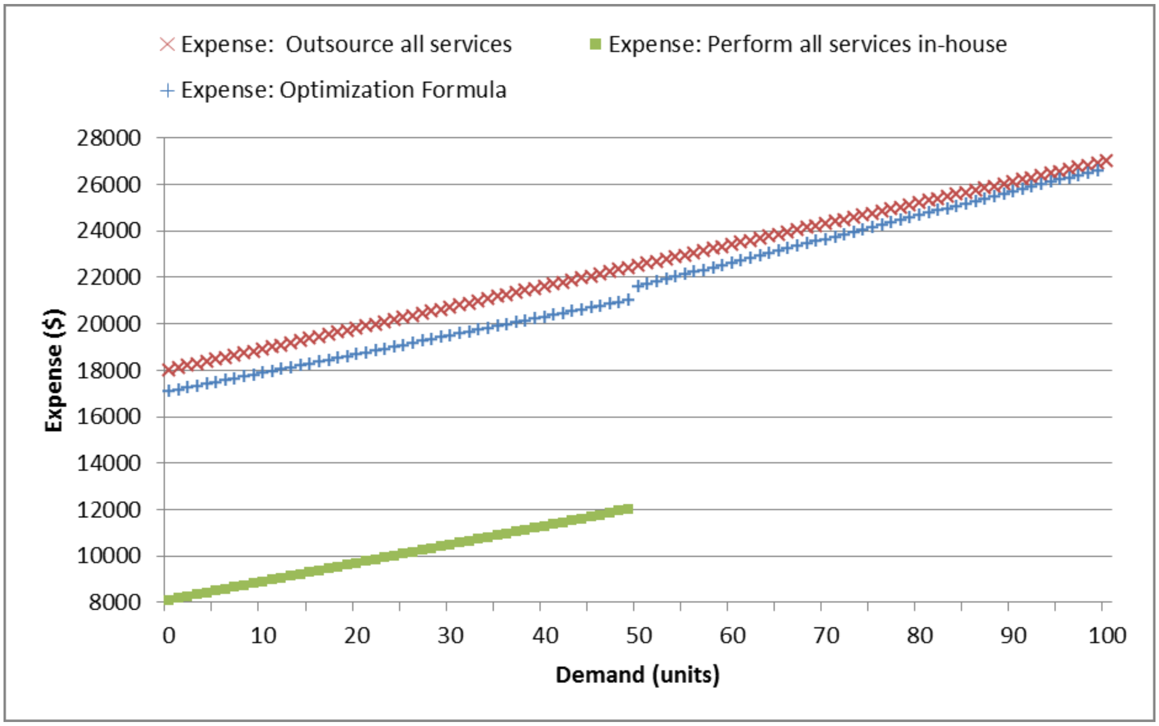


Figure 6.2: Demand vs. Expense graph for Scenario 1: Subscription Pricing Model Simulation

Figure 6.2 depicts a demand vs. expense graph that plots the results for these three cases: total expense where all services are outsourced, total expense where all services are performed in-house, and total expense using the proposed model.

Since the simulation is only varying values for x_1, the total expense for "outsource all services", depicted by x in the graph, has a baseline value of \$18,000. This baseline value is derived from the sum of the internal and external costs when $x_2=100$ and $x_3=100$.

The expense results for "perform all services in-house", depicted by ■ in the graph, have no value for x_1 greater than 50 units, since this is the point where the capacity limit is reached. This case has an expense baseline of \$8,000 which is derived from the sum of the internal costs when $y_2=50$ and $y_3=50$ which equates to $IP(y_2) + IP(y_3)$=\$8,000.

Compared to the two other cases prior to the capacity limit of 50 units, the expense is less than that of the total migration results.

The total expense for the "optimization formula" when $0 \leq d_1 \leq 50$, depicted by + in the graph, is lower than the total expense for "outsource all services" case, because the optimization function utilizes the cheapest method which is to use the internal resources first until the capacity limit is reached. The graph for this case shows two different slopes, when $0 \leq d_1 \leq 50$ and when $51 \leq d_1 \leq 100$. It is not a constant slope throughout all demands d_1 because there are added coordination costs when we mix both internal and external services when $51 \leq d_1 \leq 100$. Due to the large scale of the graph, it is not obvious that there is a \$400 difference in expense cost when using the optimization formula than that of the total migration when d_1=100. This \$400 difference is shown in Table 17.

Since there is a positive correlation between expense and quantity of x_1, the total expense is linear.

6.2.3 Demand Vs. Net Profit

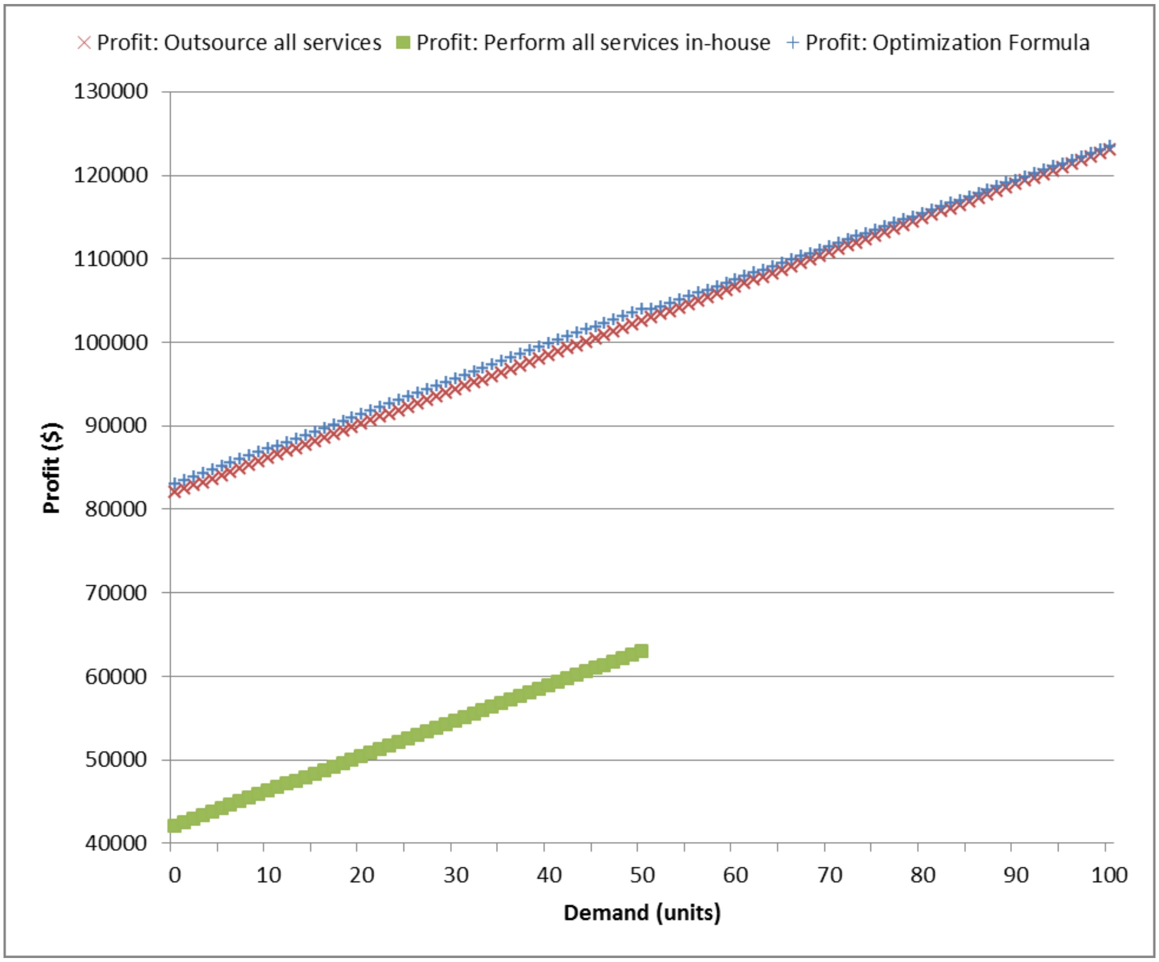


Figure 6.3: Demand vs. Profit graph for scenario 1: Subscription Pricing Model Simulation

Figure 6.3 depicts a demand vs. net profit graph that plots three results: net profits when all services are outsourced, net profits when all services are performed in-house, and net profits for the proposed model.

The trend of the profit graph for all three cases are increasing because the revenue and expense functions are both constant. From the trend of the graph, we can conclude that there is a direct correlation between the net profit and quantity of x_1.

The calculated optimal solution set, depicted by + in the graph, implies that we maximize the cheapest solution by utilizing the internal resources until the capacity limit of 50 units is reached. It is important to note the difference between *OP(x)* and *IP(x)* function is only $10 per unit. It is not obvious in the graph due to the large scale that $400 generated from the optimization formula is more than that of the total migration. This $400 difference is shown in Table 17.

6.2.4 Scenario Summary

This scenario simulated the result of a subscription-based pricing model where the annual subscription price is spread across the forecasted demands. The demand vs. revenue, demand vs. expense, and demand vs. net profit were graphed. The analysis showed the results from the use of an optimization formula yielded a larger net profit than that of the cases where all services were outsourced and where all services were performed in-house. It correctly utilized the capacity limit of the cheapest solution, which is 50 units of internal resources, before utilizing external cloud resources. In the next two sections, it outlines the results from scenarios where the revenue and cost functions are not constant.

6.3 Scenario 2: Pay-Per-Use Pricing Model Simulation

The purpose of this scenario is to show that the price can vary depending on the usage. In a variable cost (pay-per-use) pricing model for cloud computing, the usage fee is dependent on the usage.

The equations representing constant revenue function *P(d)*, linear external cost function *OP(x)*, and linear internal cost function *IP(y)* functions for this scenario are

P(d) = $500, for all demand d

$OP(x) = \$5x, \textit{ for all units of } x$

$IP(y) = \$4y + \$150, \textit{ for units of } y \leq 50$

(Where 50 units is the capacity limit for internal resources. The $150 represents the overhead costs of operating in-house.)

Table 18: Simulation results for Scenario 2: Pay-per-Use Pricing Model Simulation

Description	**Net Profit**	**Revenue**	**Net Expense**	**Coordination Costs**	**Unmet Demand**
Outsource all services	$0	$150,000	$150,000	$0	$d_1\text{-}x_1\text{-}y_1 = 0$ $d_2\text{-}x_2\text{-}y_2 = 0$ $d_3\text{-}x_3\text{-}y_3 = 0$
Perform all services in-house w.r.t. capacity limit	$22,500	$75,000	$52,500	$0	$d_1\text{-}x_1\text{-}y_1 = 50$ $d_2\text{-}x_2\text{-}y_2 = 50$ $d_3\text{-}x_3\text{-}y_3 = 50$
The proposed model	$59,442	$140,000	$80,558	$1,019	$d_1\text{-}x_1\text{-}y_1 = 7$ $d_2\text{-}x_2\text{-}y_2 = 7$ $d_3\text{-}x_3\text{-}y_3 = 6$

From the simulation in Matlab, the optimal set of x_i and y_i for this scenario is

$x_1 = 49$

$y_1 = 44$

$x_2 = 50$

$y_2 = 43$

$x_3 = 50$

$y_3 = 44$

(Note: the numbers are rounded).

The graphs illustrated in the next sections show the findings when the value of demand d_1 varies from 0 to 100.

6.3.1 Demand Vs. Revenue

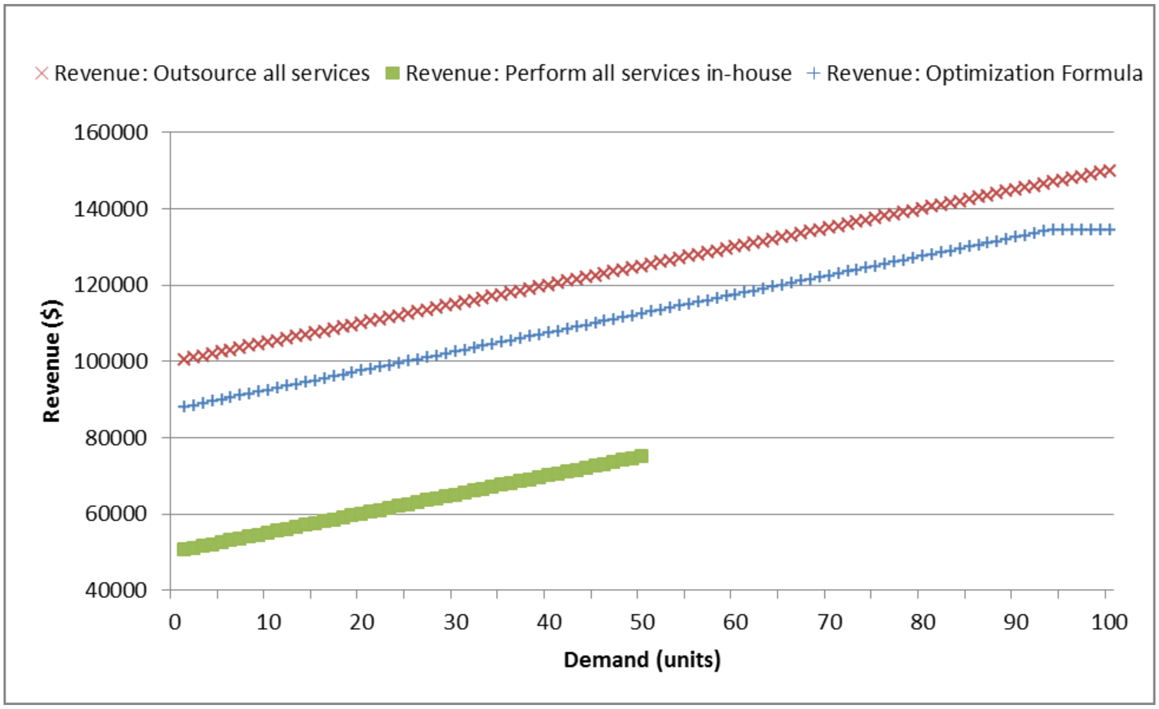


Figure 6.4: Demand vs. Revenue graph for Scenario 2: Pay-per-use Pricing Model Simulation

In this scenario the revenue results for "outsource all services" and "perform all services in-house" are the same as the previous scenario. The reason that these revenue numbers are the same since both scenarios use the same revenue function $P(d)$ = \$500.

In this scenario the optimal solution is $x_1 = 49$ and $y_1 = 44$ with an unmet demand of 7 units for service 1, 7 units for service 2, and 6 units for service 3. The optimization formula results, depicted by "+" in Figure 6.4, show that revenue plateaued after $d_1 \geq 93$ whereas the revenue for the case "outsource all services" increases at a constant rate. The reason that the revenue results plateaued after $d_1 \geq 93$ is because this is the local maximum for this scenario; by generating additional revenue, it will incur more expenses. This local maximum is depicted in the "demand vs. profit" graph.

6.3.2 Demand Vs. Expense

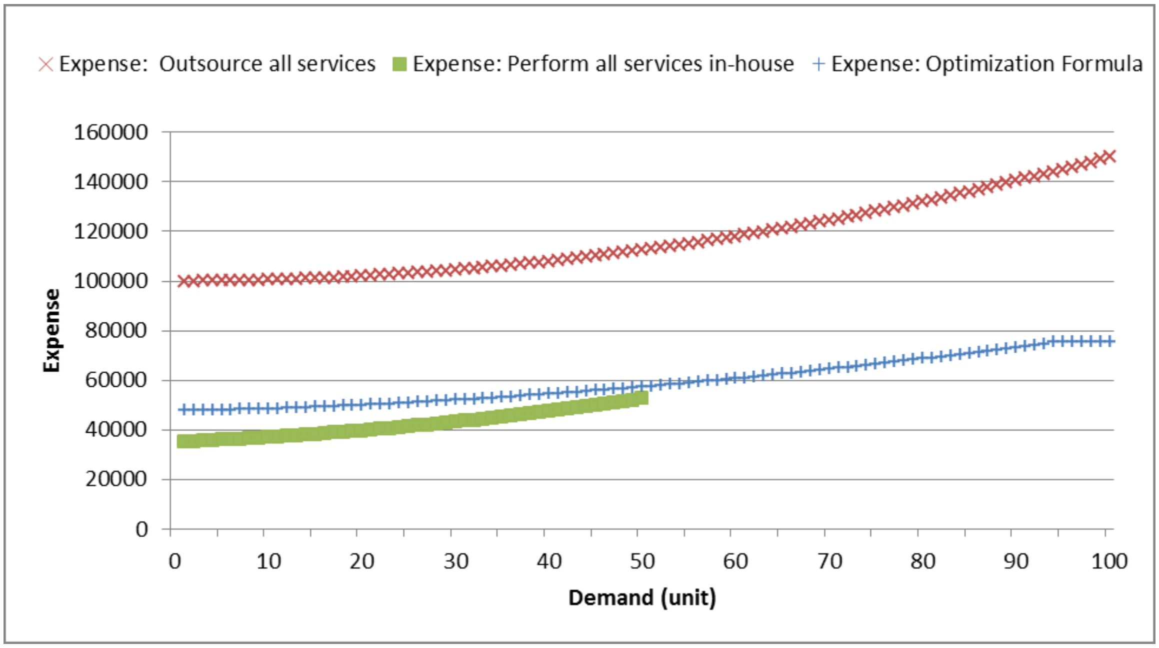


Figure 6.5: Demand vs. Expense graph for Scenario 2: Pay-per-use Pricing Model Simulation

Figure 6.5 illustrates the expense results for all three cases. Compared with the previous scenario where *OP(x)* and *IP(y)*, it is interesting to note that in this scenario the revenue results for all three cases are quadratic as opposed to linear. This is because the cost functions *OP(x)* and *IP(y)* are linear functions.

Similar to the previous scenario, the capacity limit for "perform all services in-house" is 50 units; hence, there is no value after $d_1 \geq 50$.

The results for the "optimization formula" case, depicted by "+" in Figure 6.5, show that expense plateaued after $d_1 \geq 93$ as this is the local maximum for this scenario.

6.3.3 Demand Vs. Net Profit

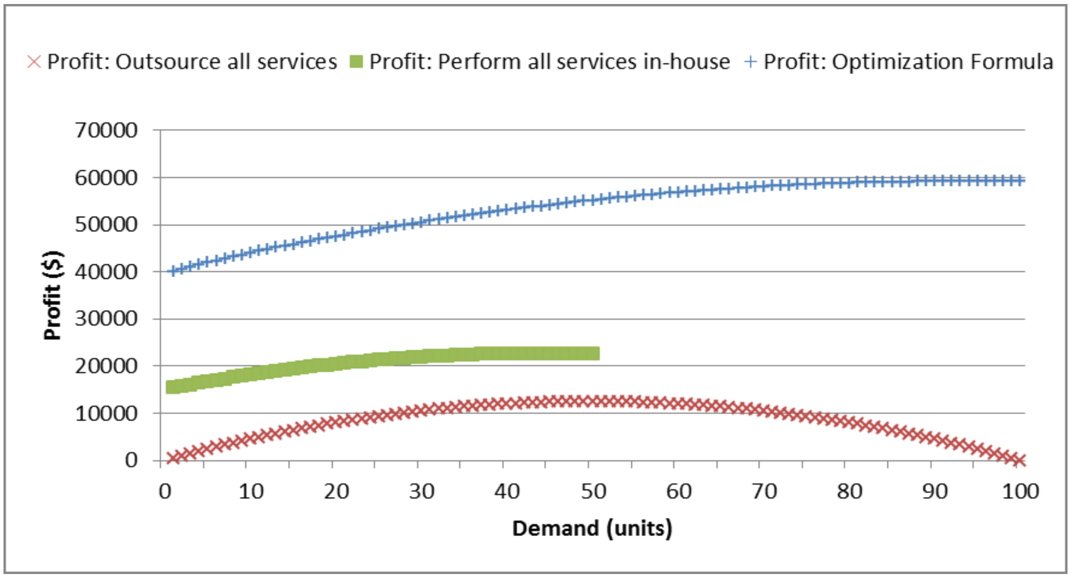


Figure 6.6: Demand vs. Net Profit graph for Scenario 2: Pay-per-use Pricing Model Simulation

Figure 6.6 illustrates the net profit results for all three cases. The graph for "outsource all services", depicted by x, is concave down which implies that there is not direct correlation between demand and profit. Unlike the previous scenario where *OP(x)* and *IP(y)* are constant functions, this scenario used linear cost functions.

In this scenario the optimal d_1 is 93 units which imply that there are 7 unmet units; as such the net-profit graph plateaued after $d_1 \geq 93$. Figure 6.6 also illustrates that the slope of the optimization formula graph is always positive up to the optimal point; the slope is 0 after the optimal point as the optimization formula has calculated that it is more profitable to have unmet demands rather than fulfilling all the requests. The unmet demands calculated by the optimization formula for this scenario are

d_1-x_1-y_1= 7

d_2-x_2-y_2 = 7

d_3-x_3-y_3 = 6.

6.3.4 Scenario Summary

This scenario used linear cost functions for *OP(x)* and *IP(y)* to simulate a pay-per-use pricing model. In this scenario we observed that the proposed model is able to determine the case where it is more profitable to have unmet demands, since fulfilling additional demands beyond this optimal point would have increased the overall expense more than that of additional revenues.

6.4 Scenario 3: Staggering Pricing Model Simulation

The purpose of this scenario is to simulate a non-linear cost using a staggering function. A staggering function is used to model situations where additional hardware is required to service additional demands. In the case of IPTV, when a single surrogate server cannot handle all the requests adding an additional surrogate server is an option, but will incur an initial fixed cost. Figure 6.7 illustrates an example of a staggering function where demand at unit 10, 20, and 40 have a sudden increase in expenses.

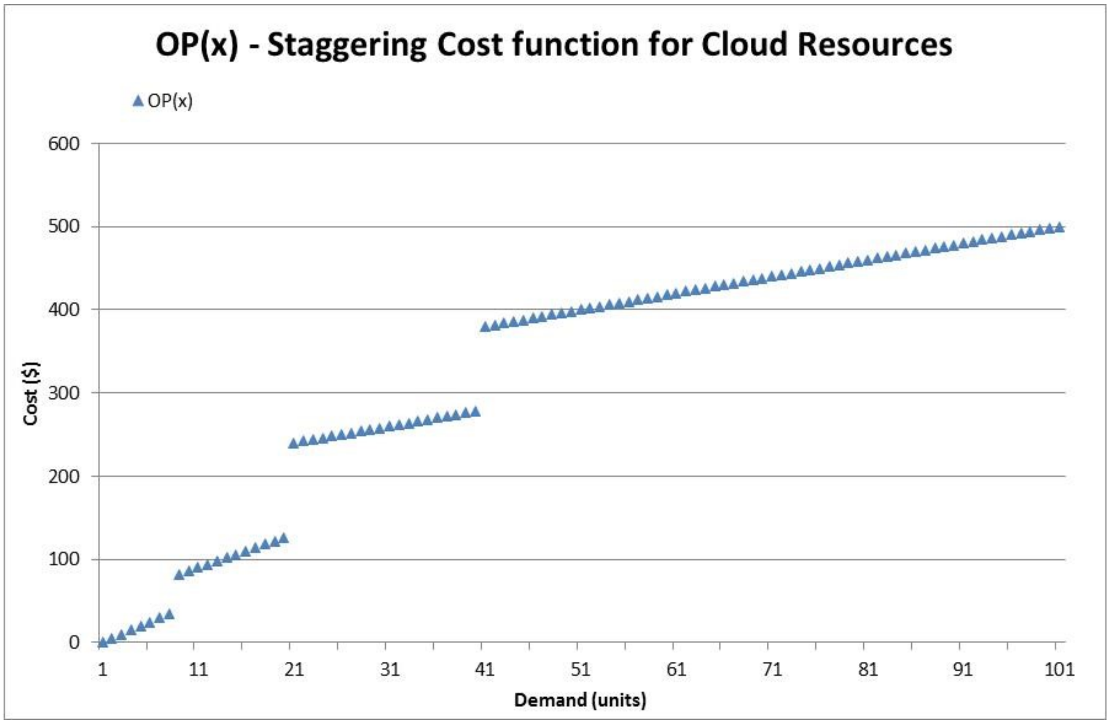


Figure 6.7: An example of a stagger-price function

The equations representing constant-revenue function *P(d)*, staggering-external-cost function *OP(x)* and linear-cost function *IP(y)* for this scenario are

P(d) = $500, for all demand d

$$OP(x) = \begin{cases} 5x, & 0 \leq x < 10 \\ 4x+50, & 10 \leq x < 20 \\ 2x+200; & 20 \leq x < 40 \\ 2x+300; & x \geq 40 \end{cases}$$

IP(y) = \$4*y*+\$150, *for units of y ≤ 50.*

Table 19: Simulation results for scenario 3: Staggering Pricing Model

Description	**Net Profit**	**Revenue**	**Net Expense**	**Coordination Costs**	**Unmet Demand**
Outsource all services	$0	$150,000	$150,000	$0	d_1-x_1-$y_1 = 0$ d_2-x_2-$y_2 = 0$ d_3-x_3-$y_3 = 0$
Perform all services in-house w.r.t. capacity limit	$22,500	$75,000	$52,500	$0	d_1-x_1-$y_1 = 50$ d_2-x_2-$y_2 = 50$ d_3-x_3-$y_3 = 50$
The proposed model	$48,038	$123,000	$74,962	$889	d_1-x_1-$y_1 = 18$ d_2-x_2-$y_2 = 18$ d_3-x_3-$y_3 = 18$

From the simulation in Matlab, the optimal set of x_i and y_i for this scenario is

$x_1 = 39$

$y_1 = 43$

$x_2 = 39$

$y_2 = 43$

$x_3 = 39$

$y_3 = 43$

(Note: the numbers are rounded).

6.4.1 Demand Vs. Revenue

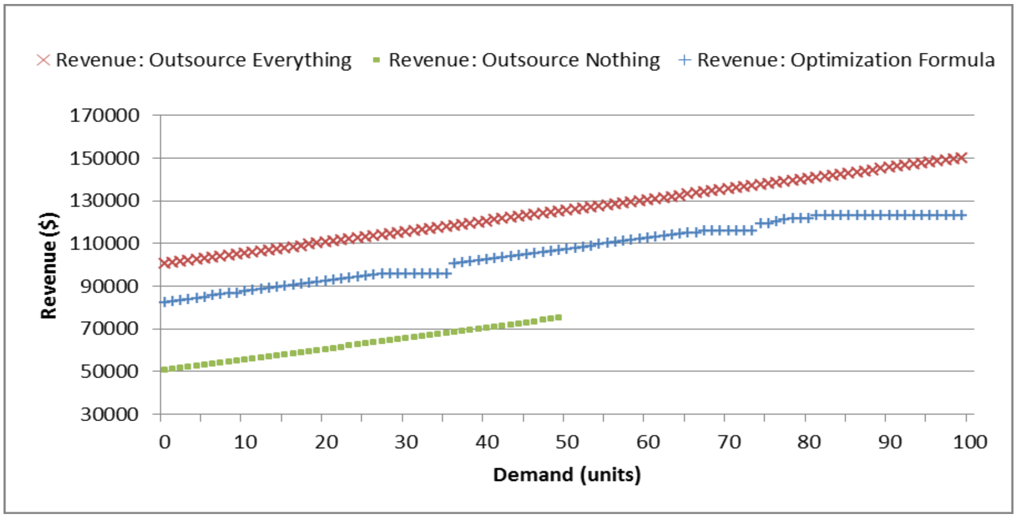


Figure 6.8: Demand vs. Revenue graph for Scenario 3: Staggering Pricing Model Simulation

Since the revenue function is constant, the revenue results for this scenario are similar to the previous scenario. In this scenario the unmet demand for service 1 is 18 units. As such, the graph for the "optimization formula", depicted by "+" in Figure 6.8, plateaued after $d_1 = 82$. It is worth mentioning that despite the "Optimization Formula" revenue results are less than that of the "Outsource Everything" results, the next two graphs show that the profit is greater.

6.4.2 Demand Vs. Expense

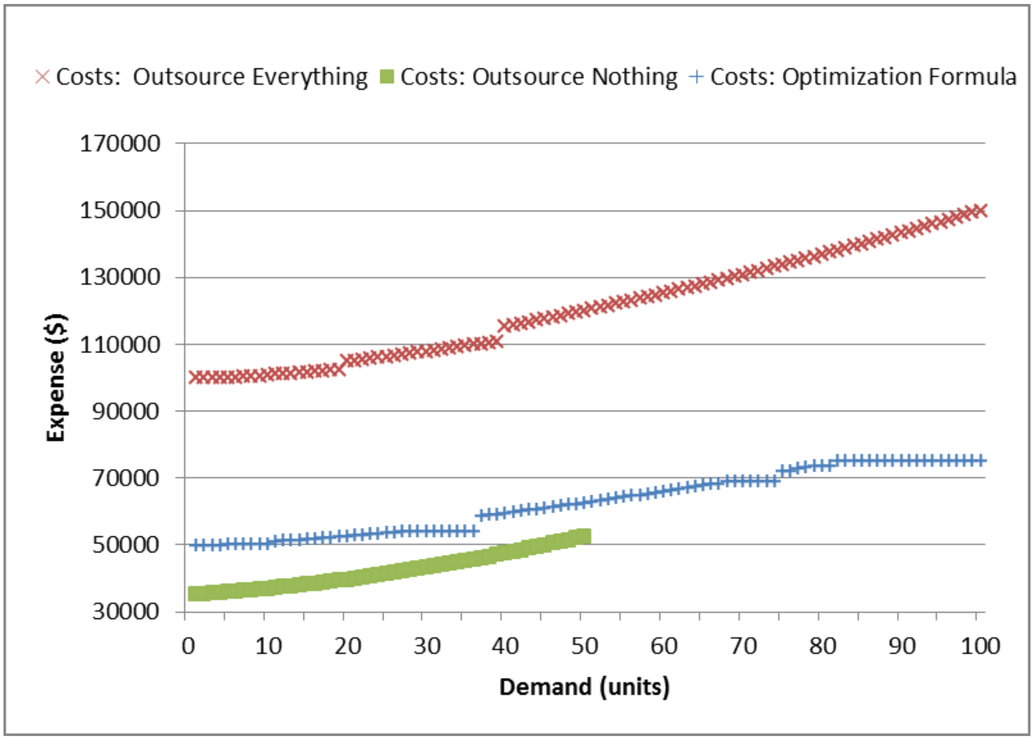


Figure 6.9: Demand vs. Expense graph for Scenario 3: Staggering Pricing Model Simulation

Figure 6.9 shows the "Demand vs. Expense" graphs for the three cases. The case where we "outsource all services", depicted by x in the graph, shows a non-continuous graph with four parts, which represent the four segments of the non-continuous *OP(x)* function.

Since the *IP(y)* function is the same as the previous scenario, the results for "perform all services in-house" is identical.

The "optimization formula" expense results, depicted by "+" in Figure 6.9, also show a non-continuous graph with four segments. The optimization formula uses a mix of internal and external resources to derive a lower expense than that of the "outsource all services" case. There are 18 units of unmet demands; consequently, the graph plateaued after $d_1 = 82$.

6.4.3 Demand Vs. Net Profit

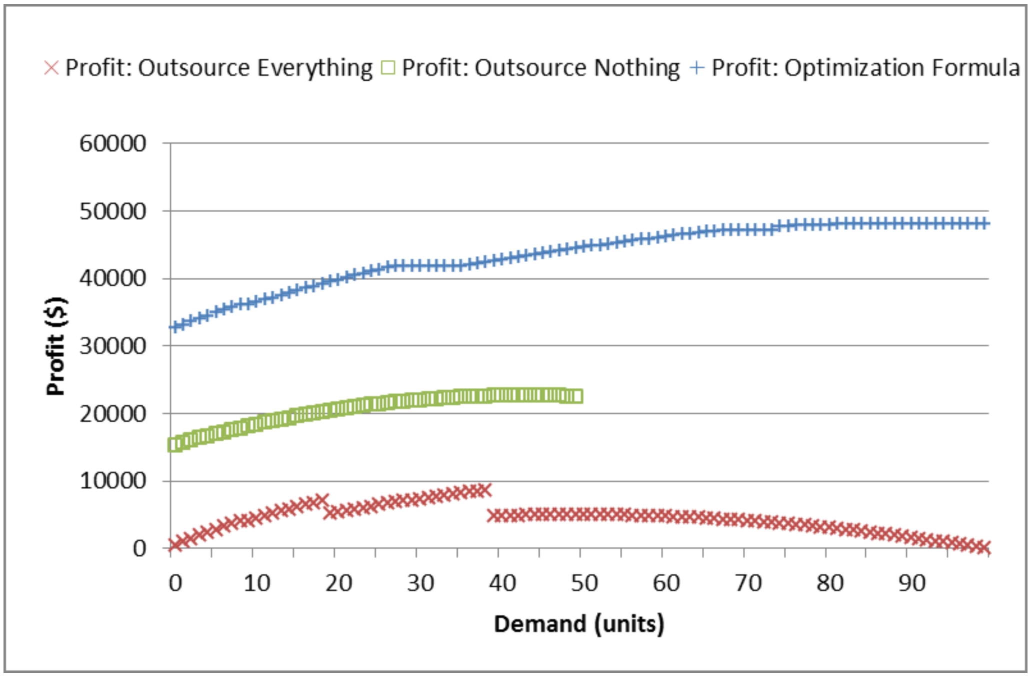


Figure 6.10: Demand vs. Net Profit graph for scenario 3: Staggering Pricing Model Simulation

Figure 6.10 shows the "Demand vs. Net Profit" graph for the three cases. Similar to the previous scenario, the "outsource all services" case, depicted by x in the graph, resulted in a concave pattern which represents that there are points in the graph where the slope is negative. When the slope is negative, the net profit is decreasing. Contrary to this case, the "optimization formula" net profit results, depicted by "+" in the graph, always has a positive slope for all demands less than the optimal solution. Once the optimal solution is reached, the net profit plateaued.

6.4.4 Scenario Summary

This scenario used a staggering cost function for *OP(x)* to simulate a staggering pricing model. This scenario validated that the proposed model is able to determine the optimal solution on a non-continuous function.

6.5 Performance Evaluation Summary

Through the three scenarios we modelled subscription, pay-per-use, and staggering pricing of the three most common cloud pricing models. Through the results from the first scenario we concluded that if an enterprise operates all of its IT services internally, it missed the opportunity to earn extra profit when the demand is beyond the capacity. From the first scenario, through the optimization-decision engine, it is able to combine a mixture of both internal and cloud resources to serve the demand to achieve a higher profit. Through the second and third scenarios, we model the cloud cost function as an increasing function based on demand. Through the last two scenarios we conclude that if an enterprise outsource all its IT services to the cloud, their profit would be less. However if an optimization-decision engine is used, it can determine an optimal set of internal and cloud IT resources and determine a local maximum for the profit function where it is more profitable to pay the SLA penalty than to acquire more cloud resources to serve more demands beyond this point. The demands beyond this local maximum is called unmet demands and it is defined in the proposed framework as d_i-x_i-y_i. The limitation of the optimization-decision engine is that it is based on maximizing the profit even if it means leaving some customer unsatisfied and may impact enterprise's reputation when unmet demands is greater than zero. This thesis acknowledge that there are some non-monetary values such as enterprise's reputation and customer satisfaction that is not easily priced to be accounted into the cost model.

Chapter 7 Conclusion and Future Work

Enterprises follows IT models, such as the Information Technology Infrastructure Library (ITIL), to define business processes that will meet their business strategies. These business processes use IT services that are subject to cloud migration. The IT service model is used as a guideline for mapping the different components of business processes. It is important to understand the interaction of business processes to IT services prior to cloud migration because there is coordination cost between IT services if different cloud vendors are used. As such, enterprises that are using the IT service model can analyze which services should be coupled together to reduce coordination costs and network delay.

When enterprises migrate to cloud services, it is important to understand the difference among cloud service models. At the Software as a Service (SaaS) model, enterprises use existing software hosted on a cloud's infrastructure to meet their business objectives. At the Platform as a Service (PaaS) model, enterprises maintain and deploy their existing software on a cloud infrastructure using the vendor's platform. At the Infrastructure as a Service (IaaS) model, enterprises maintain their software and are required to install their own platform for the software to run on; only the hardware and network components are offered by IaaS-cloud vendors. During the investigation of different cloud service models, enterprises are encouraged to compare the different pricing models offered by cloud vendors.

This thesis summarized literature relating to IT outsourcing concepts and the decision model around optimizing services to vendors. Through literature review, we outlined the similarities and differences of outsourcing and cloud computing. When enterprises leverage the use of cloud services in their business processes, they are outsourcing their IT services to cloud vendors; as such, cloud computing is a subset of outsourcing. The concept of cloud services is that enterprises are renting a service from a vendor based on the cloud-pricing model. The pricing model for cloud computing are fixed price based on a pay-per-use unit price or a subscription fee, differential price based on volume or customization of the service configuration, and market pricing based on supply and demand of vendor’s infrastructure. The pricing model affects the cost of outsourcing.

The cost of outsourcing is rolled up into three overall costs: transaction costs, IT service costs, and expected loss. Transaction costs consist of costs incurred during the planning phase which are negotiation, adoption, and allocation costs, plus costs incurred during the operational phase which are maintenance, coordination and agency costs. IT service costs are associated with the usage of the IT service. The expected loss is the cost that consists of the calculated monetary value of risks associated with confidentiality, availability, and integrity loss.

Because cloud computing has a shorter track record, companies will start by using it for less critical areas of technology before getting more comfortable with it over time. Having a framework for enterprises to analyze internal and external costs of performing their IT services is beneficial for cloud migration decision. This paper proposed a framework that enables enterprises to calculate the cost of their IT services on a cloud model to compare against the current cost (on premise or outsourced) and make decisions to increase profitability or decrease the cost of providing these services.

7.1 Summary of Contributions

A cloud computing decision framework is developed for enterprises to decompose business objectives to IT services. The proposed framework has three different levels.

During the planning level of the proposed framework, enterprises define the in-house-cost functions of operating a set of IT services in-house. By investigating different offerings from cloud vendors, enterprises can determine the external-cost function for the set of services through the use of cloud services based on usage cost. The basis of analyzing costs is to use a costing model to derive cost functions. The service-based-costing model was described to divide up the external service, software people, hardware, and facilities costs using the annual forecasted sales to calculate the unit cost for each IT service. The communication cost function is defined to determine the added cost of transferring data from one cloud service to another if they are hosted on a different vendor and from cloud service to enterprise network. The Service Level Agreement (SLA) cost function is also defined to account for penalties for not meeting the service requirement. By analyzing the business process to IT service mapping, enterprises can determine a list of services that is best to be kept together to reduce communication cost.

During the architectural level of the proposed framework, enterprises define in detail the cloud SaaS, PaaS, and IaaS services to meet the needs of the business process. At this level data models are defined to standardize the information exchange between cloud services. Once the to-be architecture is defined, enterprises can determine which IT services and their related hardware to decommission to reduce operation cost.

At the implementation level of the proposed framework, enterprises refactor their business processes to connect to cloud vendors' network to execute these cloud services. Also, enterprises can implement their monitoring services to examine that cloud services are operating within the SLA.

The contribution of this thesis is to extend a generic profit optimization decision model to cloud computing. The profit maximization formula uses the internal cost functions, and the operating costs functions for using cloud services subjected to coordination costs and SLA penalty costs to determine the percentage of services to be migrated to cloud services. The proposed profit optimization decision model takes a set of N service, revenue functions, and costs functions to compute the optimal combination of internal and external cloud services for each service in set N that will yield the maximum profit.

A simulation model is developed with different cost functions to show that the decision function is able to find the optimal percentage of services to be migrated to clouds. The simulation uses three scenarios to represent popular pricing models: constant, straight line, and staggering cost functions.

7.2 Future Work – Cloud sustainability and portability

Since cloud computing is still at its infancy, cloud vendors are offering competitive pricing models to attract enterprises to migrate their IT services to their clouds. An area to investigate further is the sustainability of cloud services. One sales strategy is that cloud vendors can operate their cloud services at a loss or near break-even for a short-term period to penetrate the enterprise market. The future work includes analyzing the impact to enterprises if a cloud vendor is bankrupted, merged with another company, or its services are obsolete.

Another area to investigate is the portability of cloud services such as how to streamline the process to switch between cloud vendors. It is more difficult to switch from one SaaS provider to another as their process and data format can be proprietary. As such, there is an opportunity for future research to standardize data format and its interoperability in SaaS using XML for enterprises to reduce the tight couplings to their SaaS vendor.

Reference

1 Grover, V., Cheon, M. J., & Teng, J. T. (1994). A descriptive study on the outsourcing of information systems functions. *Information & Management*, 27(1), 33-44.

2 Jansen, W., & Grance, T. (2011). Guidelines on security and privacy in public cloud computing. *NIST special publication*, 800-144.

3 CIO Business Technology Leadership (2010). 2011 Cloud Computing Survey. *CIO Magazine.* Retrieved 08/30, 2014, from http://mkting.cio.com/pdf/CIOCloudSummary.pdf

4 DeFelice, A., & Leon, J. C. (2010). Cloud computing: what accountants need to know. *Journal of Accountancy*, 210, 50.

5 Oredo, J. O., & Njihia, J (2014). Challenges of Cloud Computing in Business: Towards New Organizational Competencies. *International Journal of Business and Social Science,* 5(3), 150-160.

6 Martens, B., & Teuteberg, F. (2012). Decision-making in cloud computing environments: A cost and risk based approach. *Information Systems Frontiers*, 14(4), 871-893.

7 Moreira, A., Moreira, J., Sadok, D., Callado, A., Rodrigues, M., Neves, M., & Karlsson, P. P. (2011). A case for virtualization of content delivery networks. In *Proceedings of the Winter Simulation Conference*, 3183-3194.

8 Carroll, M., Van der Merwe, A., & Kotze, P. (2011, August). Secure cloud computing: Benefits, risks and controls. In *IEEE Information Security South Africa (ISSA),* 2011, 1-9.

9 Herbert, L., & Erickson, J. (2011). The ROI of cloud apps. A Total Economic Impact™ Analysis Uncovers Long-Term Value In Cloud Apps, Forrester.

10 Zuber, M. (2013). *The shocking truth about ROI and payback period in operations management*. Retrieved 08/30, 2014, from http://h30499.www3.hp.com/t5/Grounded-in-the-Cloud/The-shocking-truth-about-ROI-and-payback-period-in-operations/ba-p/6139695#.Uuq3rxCwJc4

11 Horwarth, C., Chan, W., Leung, E., Pili H. (2012). Enterprise Risk Management for Cloud computing, thoughts in ERM cloud computing. Retrieved 01/30, 2014, from http://www.coso.org/documents/Cloud%20Computing%20Thought%20Paper.pdf

12 Jadhwani, P., Mackinnon, J., & Elrefal, M. (2009). Cloud Computing-Building a Framework for Successful Transition‖. *GTSI,* Northern Virginia.

13 Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—the business perspective. *Decision Support Systems*, 51(1), 176-189.

14 Taylor, S., Cannon, D., & Wheeldon, D. (2007). ITIL: Service Operation. *TSO (The Stationery Office)*.

15 Wegmann, A., Regev, G., Garret, G. A., & Maréchal, F. (2008, September). Specifying services for ITIL service management. In *IEEE Service-Oriented Computing: Consequences for Engineering Requirements,* 2008. SOCCER'08. International Workshop, 8-14.

16 Arraj, V. (2013). Best Management Practice: ITIL the basics. Retrieved 08/30, 2014, from http://www.best-management-practice.com/gempdf/itil_the_basics.pdf

17 Morris, H., Gallacher, L. (2012). ITIL Foundation Exam Study Guide. *Sybex*. West Sussex, United Kingdom.

18 Ebert, N., Uebernickel, F., Hochstein, A., & Brenner, W. (2007). A Service Model for the Development of Management Systems for IT-enabled Services. In *AMCIS*, 455.

19 Dayal, J., Rathbone, C., Wang, L., & Von Laszewski, G. Towards On Demand IT Service Deployment, In *Internet Policies and Issues,* 7, 249-262.

20 Osterwalder, A., & Pigneur, Y. (2002). An e-business model ontology for modeling e-business. In *15th Bled Electronic Commerce Conference*, 17-19. Bled, Slovenia.

21 Osterwalder, A. (2004). The business model ontology: A proposition in a design science approach. *Institut d'Informatique et Organisation.* Lausanne, Switzerland, University of Lausanne, Ecole des Hautes Etudes Commerciales HEC, 173.

22 Franceschini, F., Galetto, M., Pignatelli, A., & Varetto, M. (2003). Outsourcing: guidelines for a structured approach. *Benchmarking: An International Journal*, 10(3), 246-260.

23 Light, B., Holland, C. P., & Wills, K. (2001). ERP and best of breed: a comparative analysis. *Business Process Management Journal*, 7(3), 216-224.

24 Kern, T., & Willcocks, L. (2002). Exploring relationships in information technology outsourcing: The interaction approach. *European Journal of Information Systems,* 11(1), 3-19. Retrieved 08/30, 2014, from http://search.proquest.com/docview/218781511?accountid=14701

25 Han, H. S., Lee, J. N., & Seo, Y. W. (2008). Analyzing the impact of a firm's capability on outsourcing success: A process perspective. *Information & Management*, 45(1), 31-42.

26 Blindenbach-Driessen, F., & van den Ende, J. (2006). Innovation in project-based firms: The context dependency of success factors. *Research Policy*, 35(4), 545-561.

27 Prencipe, A., & Tell, F. (2001). Inter-project learning: processes and outcomes of knowledge codification in project-based firms. *Research policy*, 30(9), 1373-1394.

28 Gilley, K. M., & Rasheed, A. (2000). Making more by doing less: an analysis of outsourcing and its effects on firm performance. *Journal of Management*, 26(4), 763-790.

29 Akbar, R., & Hassan, M. F. (2010). Limitations and measures in outsourcing projects to geographically distributed offshore teams. In *IEEE Information Technology (ITSim), 2010 International Symposium,* 3, 1581-1585.

30 P. Mell, T. Grance, (2010). The NIST definition of cloud computing. *NIST*. Retrieved 08/30, 2014, from http://www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf

31 Archer, J., & Boehm, A. (2009). Security guidance for critical areas of focus in cloud computing. *Cloud Security Alliance*, 2, 1-76.

32 Cusumano, M. (2010). Cloud computing and SaaS as new computing platforms. *Communications of the ACM*, 53(4), 27-29.

33 Chang, V., Bacigalupo, D., Wills, G., & De Roure, D. (2010). A categorisation of cloud computing business models. In *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing,* 509-512.

34 Indla, Y., & Kumar, M. S. (2012). Extended Group Key Transfer Protocol for Authentication Using DES based on Secret Sharing in Cloud. *International Journal of Emerging Technology and Advanced Engineering*, 2(11), 541-551.

35 Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, *39*(1), 50-55.

36 Sun Microsystems Inc. (2010). Introduction to cloud computing architecture [White Paper]. Retrieved 08/30, 2014, from http://eresearch.wiki.otago.ac.nz/images/7/75/Cloudcomputing.pdf

37 Datatrak. (2012). Understanding the cloud. Retrieved 08/30, 2014, from http://www.slideshare.net/DATATRAK/understanding-thecloud

38 Hurwitz, J., Bloor, R., Kaufman, M., & Halper, F. (2009). *Cloud computing for dummies,* 1. John Wiley & Sons.

39 QSAR World. (n.d.). Cloud Computing. Retrieved 01/30, 2014, from http://www.qsarworld.com/files/Cloud-computing.pdf

40 Ehrenhofer, C., & Kreuzer, E. (2012). The Role of Business Model Design in the Service Engineering Process: A Comparative Case Study in the Field of Cloud Computing to Join Service Engineering with Business Model Design. In IEEE *SRII Global Conference (SRII),* 283-292.

41 Hajjat, M., Sun, X., Sung, Y. W. E., Maltz, D., Rao, S., Sripanidkulchai, K., & Tawarmalani, M. (2010). Cloudward bound: planning for beneficial migration of enterprise applications to the cloud. *ACM SIGCOMM Computer Communication Review*, *40*(4), 243-254.

42 Harmon, R., Demirkan, H., Hefley, B. & Auseklies, N. (2009). Pricing Strategies for Information Technology Services: A Value-Based Approach. Proceedings of *42nd Hawaii International Conference on System Sciences*, 1-10.

43 Jäätmaa, J. (2010). Financial aspects of cloud computing business models. *Espoo: Aalto University*.

44 Amazon Web Services. Amazon EC2 Pricing. Retrieved 08/30, 2014, from http://aws.amazon.com/ec2/pricing/

45 Amazon Web Services. Amazon EC2 Spot Instance. Retrieved 08/30, 2014, from http://aws.amazon.com/ec2/spot-instances/#6

46 Baker, C. (2013). Cloud computing and outsourcing: Where does one end and the other begin? *ZDNet*. Retrieved 02/22, 2013, from http://www.zdnet.com/cloud-computing-and-outsourcing-where-does-one-end-and-the-other-begin-7000017827/

47 Dhar, S. (2012). From outsourcing to Cloud computing: evolution of IT services. *Management Research Review*, 35(8), 664-675.

48 Bender, D. (2012). Privacy and Security Issues in Cloud Computing. *The Computer & Internet Lawyer*, 29(10), 1-15.

49 Rose, K. H. (2013). A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition. *Project Management Journal*, 44(3).

50 Sounderpandian, J., Balakrishnan, J., Cheng, C. H., & Wong, W. C. (2010). A quantitative model for understanding multiple vendor IT outsourcing. *International Journal of Procurement Management*, 3(4), 428-450.

51 Meskill, B., Davy, A., & Jennings, B. (2012). Revenue-maximizing server selection and admission control for IPTV content servers using available bandwidth estimates. In IEEE *Network Operations and Management Symposium (NOMS),* 319-326.

52 Tran, V., Keung, J., Liu, A., & Fekete, A. (2011). Application migration to cloud: a taxonomy of critical factors. In *Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing*, 22-28.

53 L. Columbus (2012). First Steps to Creating a Cloud Computing Strategy for 2013. *Forbes*. Available at: http://www.forbes.com/sites/louiscolumbus/2012/12/18/first-steps-to-creating-a-cloud-computing-strategy-for-2013/

54 Google (2014). Google Maps API for Business. Retrieved 08/30, 2014, http://www.google.ca/enterprise/mapsearth/products/mapsapi.html

55 Litoiu, M., Woodside, M., Wong, J., Ng, J., & Iszlai, G. (2010). A business driven cloud optimization architecture. In *Proceedings of the 2010 ACM Symposium on Applied Computing,* 380-385.

56 Liu F., Tong J., Mao J., Bohn R., Messina J., Badger L., & Leaf D (2011). NIST Cloud Computing Reference Architecture. *NIST special publication,* 500, 292.

57 Gonidis, F., Paraskakis, I., & Kourtesis, D. (2012). Addressing the challenge of application portability in cloud platforms. In *7th South East European Doctoral Student Conference*, 565-576.

58 Pink Elephant (2005). Defining, modeling & costing IT services. Retrieved 01/30, 2014, from https://www.pinkelephant.com/DocumentLibrary/UploadedContents/PinkLinkArticles/Defining%20Modeling%20and%20Costing%20IT%20Services%20-%20Final.pdf

59 Lee, J. N., Huynh, M. Q., Chi-wai, K. R., & Pi, S. M. (2000, January). The evolution of outsourcing research: what is the next issue? In IEEE *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference*, 1-10.

60 Pinnington, A., & Woolcock, P. (1995). How far is IS/IT outsourcing enabling new organizational structure and competences? *International Journal of Information Management*, 15(5), 353-365.

61 Durkee, D. (2010). Why cloud computing will never be free. *Queue*, 8(4), 20.

62 Asit, D., Davis, D., Kearney, R., Keller, A., King, R., Kuebler, D., Ludwig, H., Polan M., Spreitzer M., & Youssef, A. (2004). Web services on demand: WSLA-driven automated management.*IBM systems journal*, 43(1), 136-158.

63 Bisong A, Rahman S: An overview of the Security concerns in Enterprise Cloud Computing. *International Journal of Network Security & Its Applications (IJNSA) 2011*, 3(1), 30–45.

64 Rackspace Support (2013). Understanding the Cloud Computing Stack: SaaS, PaaS, IaaS. October 2013. Retrieved 08/30, 2014, from http://www.rackspace.com/knowledge_center/whitepaper/understanding-the-cloud-computing-stack-saas-paas-iaas

65 Nazari-Shirkouhi, S., Eivazy, H., Ghodsi, R., Rezaie, K., & Atashpaz-Gargari, E. (2010). Solving the integrated product mix-outsourcing problem using the imperialist competitive algorithm. *Expert Systems with Applications*, *37*(12), 7615-7626.

66 Bellefeuille, C. L., Quantifying and Managing the Risk of Information Security Breaches to the Supply Chain. *Massachusetts Institute of Technology*, 2005. Retrieved 08/30, 2014, from http://dspace.mit.edu/bitstream/handle/1721.1/33313/62311875.pdf?sequence=1

67 Menai, M. F., Fieau, F., Souk, A., & Jaworski, S. (2009). Demonstration of standard IPTV content delivery network architecture interfaces: prototype of standardized IPTV unicast content delivery server selection mechanisms. In IEEE *Consumer Communications and Networking Conference, 2009*, 6, 1-2.

68 Kandavanam, G., Botvich, D., & Balasubramaniam, S. (2010). PaCRA: A path-aware content replication approach to support QoS guaranteed video on demand service in metropolitan IPTV networks. In IEEE *Network Operations and Management Symposium (NOMS),* 2010*,* 591-598).

69 Pallis, G., & Vakali, A. (2006). Insight and perspectives for content delivery networks. Communications of the ACM, 49(1), 101-106.

70 Blair, A., Parr, G., Morrow, P., Scotney, B., McConnell, A., Appleby, S., & Nilsson, M. (2012). Cloud based Dynamically Provisioned Multimedia Delivery: An Elastic Video Endpoint (EVE). In *CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization*, 260-265.

71 Passarella, A. (2012). A survey on content-centric technologies for the current Internet: CDN and P2P solutions. Computer Communications, 35(1), 1-32.

72 Díaz, M. E. A., & Huh, E. N. (2010). An efficient IPTV cost-based policy management of hosting service for 3rd party content. In IEEE *Ubiquitous Information Technologies and Applications (CUTE), 2010 Proceedings of the 5th International Conference*, 1-6).

73 Popescu, A., Erman, D., Ilie, D., Constantinescu, D., & Popescu, A. (2006). Internet Content Distribution: Developments and Challenges. In *4th Swedish National Computer Networking Workshop SNCNW2006*.

74 Blair, A., Parr, G., Morrow, P., Scotney, B., McConnell, A., Appleby, S., & Nilsson, M. (2012). Cloud based Dynamically Provisioned Multimedia Delivery: An Elastic Video Endpoint (EVE). In *CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization*, 260-265.

75 Panwalkar, S. S., Smith, M. L., & Koulamas, C. P. (1993). A heuristic for the single machine tardiness problem. *European Journal of Operational Research*, *70*(3), 304-310.