Uncertainty and Uncertainty Tolerance in Service Provisioning

Thesis by
Johari Abdullah

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

Newcastle University
Newcastle Upon Tyne, UK

March 2014

To maria, aleesha, and atiya...

Acknowledgements

This 4 years period of study has been a long journey with many trials and tribulations. There have been ups and downs, roadblocks, and dead ends. But at the same time, I have also gained various skills and knowledge during this period. All these are not possible without the help, guidance, and support of various people during my time here at Newcastle University, UK.

Firstly, I would like to express my gratitude to my supervisor, Prof. Aad van Moorsel for his valuable guidance and support throughout the duration of my study here at Newcastle University. His comments and insight during our various discussion sessions help me to shape my research and solve many research problems. Special thanks for Dr. Graham Morgan and Dr. Steve Riddle for being members of the supervision committee.

I would also like to acknowledge and express my gratitude to UNIMAS (Universiti Malaysia Sarawak) and the Government of Malaysia for the opportunity and sponsorship of my study.

During the period of my study here at Newcastle University, I have enjoyed the company of many colleagues and friends: Haryani, Hisyam, Kamarul, Hanani, Nazim, Yusnaidi, Ayyad, Osama, and AbuBkr. I enjoyed the discussion sessions, general jokes and banters, and also the family outings.

Finally, I do not have words to express my gratitude to my family: my beloved other half, *maria* who always provide support and love to me, and my two daughters, *aleesha* and *atiya* who bring joy and smile at all times. To my mother, Normah, who have given love and support throughout my life, thank you for everything.

Abstract

Service, in general term is a type of economic activity where the consumers utilize labour and/or expertise of others to perform a specific task. The birth and continued growth of the Internet provide a new medium for services to be delivered, and enable services to become widely and readily available. In recent years, the Internet has become an important platform to provide services to the end users. Service provisioning, in the context of computing, is the process of providing users with access to data and technology resources. In a perfect operating environment, the entities involved can expect the system will perform as intended or up to an accepted level of quality. Unfortunately, disruptions or failures can occur which can affect the operation of the service. Thus, the entities involved, in particular the service requester faces a situation whereby the service requester's belief towards certain process in the service provisioning lifecycle is affected, i.e. deviates from the actual truth. This situation whereby the service requester's belief is affected is referred as an uncertainty.

In this thesis, we discuss and explore the issue of uncertainty throughout the service provisioning lifecycle and provide a measure to tolerate uncertainty in service provisioning offer through the application of subjective probability framework. This thesis provides several key contributions to address the uncertainty issues in service provisioning system in particular, for a service requester to overcome the negative consequence of uncertainty. The key contributions are: (1) introduction to the issue of uncertainty in service provisioning system, (2) a new classification scheme for uncertainties in service provisioning system, (3) a unified view of uncertainty in service provisioning system based on temporal classification, which is linked to service requester's view, (4) a concept of uncertainty tolerance for service provisioning, (5) an approach and framework for automated uncertainty tolerance in service provisioning offer.

The approach and framework for uncertainty tolerance in service provisioning offer presented in this thesis is evaluated through an empirical study. The result from the study shows the viability of the approach and framework of the uncertainty tolerance mechanism through the application of subjective probability theory. The result also shows the positive outcome of the mechanism in term of higher cumulative utility, and better acceptance rate for the service requester.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis contributes to the area of Services Computing by proposing an automated solution for addressing the problem of uncertainty in a service provisioning system. Uncertainty and service are two important concepts that we encounter in our daily life. We experience uncertainty in multitude of situations and aspects of life. For example, uncertainty in simple routines like choosing attire to work (What to wear this morning?), weather condition (Is it going to rain in the afternoon?), traffic condition (whether the route that I always take in the morning is going to be busy with traffic) and in much more important situation such as economic and political uncertainty. All these situations involve uncertainty, which is due to various factors such as variability, imprecision, and in most cases is due to the lack of information.

Service, in general term is a type of economic activity where the consumers consume or utilize labour and/or expertise of others to perform a specific task. For example, when a person use a postal delivery service to send a letter, that person is using an organized system of labour and materials to accomplish that task. If someone visit a doctor for a consultation, that person is utilizing a service in the form of expertise. Similar to uncertainty, we utilizes (or provide) service throughout our daily life. Services are two key components of economics, the other being goods. Several researches have shown the shift from economy based on goods towards economy based in services in many countries [3, 4].

The birth and continued growth of the Internet provide a new medium for services to be delivered. The Internet, which eliminate physical and geographical boundaries and limitation enable services to become widely and readily available to the general public and also organization and businesses. In recent years, the Internet has become an important platform to provide services to the end users. Services Computing[5], Service-oriented Computing[6], Cloud Computing[7], Web Services[8] and Utility Computing[9] are all different paradigm of implementation for distributed systems[10]. They all share similar objective, which is to provision services in electronic form. Provisioning, in the context of computing, is the process of providing users with access to data and technology resources.

1.1 Motivation

Service-oriented computing (SOC) is defined as a computing concept that employs services as the building block for developing applications [11]. SOC (in the form of grid computing) has evolved from an economic perspective, where it shifted from merely being the tools for academics and research (lack notion of services for profit) [12, 13], to the latest paradigm in the form of cloud computing which is geared towards corporate, enterprise and general public [14]. Latest statistics from Netmetix [15] indicates that the global cloud computing market is predicted to rise from $40.7 billion to $241 billion in 2020 and Amazon, one of the key service provider in cloud computing market is expected to make $750 million profit in 2011 from its Amazon Web Services (AWS) offering. These two statistics indicate the growing economics importance of service-oriented computing.

Service provisioning consists of several steps or activities between several entities, mainly the service requester and service provider. In a perfect operating environment, the entities involved can expect that the system will perform as intended or up to an accepted level of quality. Unfortunately, disruptions or failures can occur which can affect the operation of the service. Thus, the entities involved, in particular the service requester faces a situation whereby the service requester's belief towards certain process in the service provisioning lifecycle is affected, i.e. deviates from the actual truth. In the above example, this situation whereby the service requester's belief is affected is referred as an *uncertainty*. In this thesis, we discuss and explore the issue of uncertainty throughout a service provisioning lifecycle, and we are interested to address the issues related to uncertainty in term of the service requester's belief.

One of the challenges in service-oriented system is to provide and guarantee an agreed level of service to the end users. Since services have no notion of tangibility (physical cue such as colour, smell, etc) which can be used by customers to evaluate the quality of service, service providers need a mechanism to indicate to potential customers their commitment (to service quality).

The term Quality of Service (QoS), is generally defined as *"The degree to which a provided activity promotes customer satisfaction"* [16]. From the point of view from the field of networking and telephony, as defined by ITU-T [17], QoS refers to a set or collection of performance metrics that determines the degree of satisfaction of the user. The goal of QoS is to provide preferential service to the end users based on the needs of specific application such as voice over IP (VoIP), multimedia streaming, etc. QoS is a way for the service provider to establish the level of quality provided and usually dictates the cost of the service based on the QoS provided. Unfortunately, QoS is just a guideline and is not bound by any contractual agreement between a service provider and the end users, as compared to Service Level Agreement which is a form of contractual agreement. Therefore, within the context of this research, in theory, uncertainty issue can also be applied to the QoS. However, this research focuses on the issue of uncertainty within the scope of Service Level Agreement in the form of contractual agreement between the service provider and the end user.

One way to convey the confidence (in service quality) to the customers is by employing a form of contractual agreement such as Service Level Agreement (SLA) between the service providers and the customers. An SLA is a part of a service contract where the level of service is formally defined between the consumer (end user) and the service provider [18]. The idea of having a contract between two parties to guarantee service quality has been proposed and used for IP-based network [19] and also Next Generation Network [20]. The same concept of quality guarantee is applied to the different paradigm of service-oriented system such as grid computing, cloud computing, and web services.

In a commercial provisioning environment, SLA can be a key factor in attracting potential consumers [21]. For example, service providers that can provide a guaranteed quality of service will more likely be chosen by a customer. Furthermore, if the service being provisioned is used by consumers to operate their own business operation, the quality and guarantee of the service offer becomes important[22].

1.2 Research Problems

The general research problem is stated as follows:

> *How can we provide an automated solution to overcome or reduce the problem of uncertainty in a service provisioning environment?*

The above general research problem can be subdivided into four specific research questions:

> **RQ1:** What is the definition and argument about uncertainty, specific to the area of service provisioning?

To date, there has been no attempt to formally discuss, and define the issue of uncertainty in service provisioning. The lack of consistent definition and discussion resulted in multitude of different interpretation of uncertainty, and also lack of focus for researchers whom might be investigating the same problem.

> **RQ2:** Given that there are uncertainties in service provisioning, are they all the same? How to classify these uncertainties according to specific criteria?

A service provisioning lifecycle contains several processes that might have associated uncertainty. These uncertainties can be of different types, therefore need to be classified accordingly. The lack of consistent classification scheme can cause two problems: (1) since the type of uncertainty can be linked to different treatment or solution, the lack of proper classification can lead to unsuitable treatment (less efficient or accurate), (2) lack of consistent classification can lead to poor understanding of the underlying problem. Furthermore, there is no comprehensive view on the issue of uncertainty in service provisioning. There are different types of uncertainty and uncertainty can exists in different phases of the service provisioning lifecycle. The lack of a consistent view of uncertainty in service provisioning can leads to confusion among interested parties.

> **RQ3:** Which technique or approach is suitable to provide uncertainty tolerance for uncertainty in service provisioning?

Given the subjective nature of uncertainty in service provisioning system, and multiple potential solution, there is a need to select the most suitable approach which can involve multiple domains such as subjective probability, economics, and decision theory. Such technique or approach should be able to reduce the negative effect of the uncertainty, and facilitates customer's belief adjustment.

> **RQ4:** Can we devise a framework to support automated uncertainty tolerance mechanism in a service provisioning system?

Given a service provisioning system with associated uncertainty problem, there is a need for a framework that can facilitates the design and development of the uncertainty tolerance mechanism that can be part of a service provisioning system. The components within the framework, especially the uncertainty tolerance mechanism should be automated as much as possible in order to minimize customers intervention and enable efficient process.

> **RQ5:** Given the above approach and framework, how do we evaluate the viability and effectiveness of the proposed solution?

Finally, after designing and implementing the proposed solution (based on the approach and the framework), there is a need for an empirical study to validate and evaluate the solution. The result from this study will help us to understand the strength and probable limitation of our solution.

1.3 Summary of Contributions

This thesis addresses the issue of uncertainty in a service provisioning environment in general and focuses on the issue of uncertainty in selected phase of a service provisioning lifecycle. The key area of concern is the phase when a customer is presented with the service guarantee offer and subsequently has to make decision whether to accept or reject the service based on the offer.

This thesis provides the following four key contributions to address the issue of uncertainty in service provisioning.

- An introduction to the issue of uncertainty in service provisioning, which includes (i) a definition of uncertainty in service provisioning, (ii) a new hybrid classification scheme, and (iii) a consistent view of uncertainty. The resulting hybrid classification scheme is based on several dimensions and is useful to interested parties such as researchers, system designer, and system developer as a tool to classify uncertainty in a service provisioning system. The view of uncertainty is based on the temporal classification scheme and service requester's perspective. This view is important and useful since it provides a consistent and clear understanding of the problem of uncertainty in service provisioning. Having such a view enables various interested parties (such as researchers, system designer, and system developer) to work from the same reference point that is as unambiguous as possible.

- A generic concept of uncertainty tolerance for uncertainty in service provisioning. This includes the definition of uncertainty tolerance, a discussion on the needs for uncertainty tolerance, and the means on how to achieve uncertainty tolerance. The proposed concept is based on the temporal view of service provisioning lifecycle and service requester's belief. This concept outlines the requirements to tolerate the uncertainty in service provisioning but does not enforce any specific approach, thus enabling interested parties to find the best suitable approach for tolerating uncertainty.

- An approach to uncertainty tolerance for service provisioning offer. The proposed approach utilizes subjective probability framework to tolerate the uncertainty. One key element which differentiates our approach to existing works is that our approach allows a service requester to express or assigns initial belief towards a service offer from a service provider. This approach also includes the application of expected utility theory, which facilitates the decision making process of the service requester.

- A framework for an automated uncertainty tolerance in service provisioning. The proposed framework is designed based on the underlying principles of subjective probability theory in tolerating the uncertainty and provides the operationalization details of the underlying theory. The framework provides the processing logic and semantic of the uncertainty tolerance engine. The framework will be useful to system designer and service provider in designing a service provisioning system that takes into account the problem of uncertainty and tolerates the uncertainty accordingly.

1.4 Publication

This thesis includes work that has been published in peer-reviewed workshop, conference, and journal as follows:

- Johari Abdullah and Aad van Moorsel. **Uncertainty and Uncertainty Tolerance in Service Provisioning**. *Journal of Internet Services and Information Security (JISIS)*, 1(4):89109, 11 2011. [23]

- Johari Abdullah and Aad van Moorsel. **Uncertainty and Uncertainty Tolerance in Service Provisioning**. INTRUSO2011 (1st INternational Workshop on TRUstworthy Service-Oriented Computing. Copenhagen, June 27-28, 2011, Technical University of Denmark.

1.5 Structure of Thesis

The remainder of the thesis is structured as follows:

Chapter 2 presents the closely related works that addresses the problem of uncertainty in computing related areas.

Chapter 3 discusses the background of two important concepts in this thesis; (1) services, and (2) uncertainty. For services, we discuss and define the term service and service provisioning and look at service concept from economics point of view, comparing to the concept of goods. As for the issue of uncertainty, we look at a wide variety of other research areas to get an insight on two matters, in which how different fields: (1) define the term uncertainty, and (2) classify uncertainty.

Chapter 4 contains three important sections. Firstly, the chapter presents a unique classification scheme for uncertainty in service provisioning using different dimensions. Secondly, it presents a new view of uncertainty using the temporal classification scheme and the service requester's perspective. Finally, the chapter introduces the generic concept of uncertainty tolerance in service provisioning.

Chapter 5 presents an approach to tolerate uncertainty for a specific case of uncertainty in the service offer from a service provider. This approach utilizes the generic concepts of uncertainty tolerance presented in Chapter 4, and employs subjective probability framework, which includes evidence gathering approach, and also expected utility theory to assist service requester in decision making.

Chapter 6 focuses on the development of an automated framework for tolerating uncertainty in service provisioning, specifically for the service provisioning offer. The aims of the framework is to provide an application level blueprint for interested parties to implement the uncertainty tolerance mechanism in their service-based system.

Chapter 7 provides an empirical study to validate the viability of the uncertainty tolerance concept discussed in Chapter 4, the approach for uncertainty tolerance presented in Chapter 5, and the framework presented in Chapter 6. Subsequently, the results and findings of the study is also discussed.

Chapter 8 provides a summary of the research, and restates the major contributions and findings. This chapter also discusses issues that require further research.

Chapter 2

Literature Review

2.1 Introduction

This chapter presents a summary of related work forming a background for the research work presented in this thesis. The relevant literature pertaining to the importance of the research question and available solutions is explored and reviewed.

2.2 Uncertainty in Services Computing

The issue of uncertainty is not something new in many research fields but surprisingly, there is minimal research conducted on uncertainty issues related to the area of service computing. To date, there has been no specific research works done to address in general the issue of uncertainty in the area of service provisioning. The lack of research leads to no proper definition of what uncertainty is (with respect to service provisioning), and no classification scheme to classify uncertainty into a specific classes or categories. Perhaps this is not surprising due to the fact that there exists different definition and interpretation of uncertainty in other research areas.

The lack of specific definition and classification of uncertainty in service provisioning resulting in the lack of focused and concerted effort within services computing research area. Hence, there might be other research working on solving the problem of uncertainty without specifically acknowledging that the crux of the problem they trying to solve is related to uncertainty (in service provisioning). Therefore, Chapter 3 will provide in-depth review of the issues of uncertainty in other fields, including the definition and classification of uncertainty, which leads to a specific definition and classification of uncertainty in service provisioning.

Based on our literature survey, there has been no effort in the past to specifically consider uncertainty as the main focus of research problem within service provisioning system. Existing research works in the past, in general, focused on two main issues which we believe has the root of the problem linked to uncertainty. These two issues are (1) quality uncertainty, and (2) trust and uncertainty.

2.2.1 Quality Uncertainty

The issue of uncertainty is a common problem in IT related services such as multimedia streaming, VOIP (voice over IP), online trading systems and customer service call centres [24]. This uncertainty can negatively affect either the service provider or service requester [25]. One of our contribution of this research is to address the issue of compliance of Service Level Agreement, when the customer has imperfect belief (due to gap in knowledge), which falls under the quality uncertainty scope of research. A recent paper by Smith in 2010 [26] looks into the issue of uncertainty in service provision contracts. The main objective of Smith's research is to look into the issue of service provider ability to fulfil the service quality guarantees due to non-deterministic uncertainty such as fluctuation in network/server loads and hardware failures and provide a mechanism to optimize service provision contract under this uncertainty. Smith's approach to solve the uncertainty issue is to use utility model. There are several differences between Smith's work and what is presented in this thesis. Firstly, there is no attempt or reference to the issue of uncertainty in general within the context of service provisioning since Smith's paper focuses specifically on the uncertainty issue from the service provider point of view. Our research provides, firstly, the different type of uncertainty in a complete lifecycle of a service provisioning process, and secondly provide a classification scheme that enable the grouping of uncertainty based on several specific dimensions or characteristics. Secondly, instead of focusing on the uncertainty issue from the perspective of service provider, we have taken the initiative to look into the issue of uncertainty from customer point of view, and provide a mechanism that tolerate the uncertainty, and assist customer in decision making. In a way, this research work is complementary to the work conducted by Smith. Finally, similar to Smith's approach in mitigating uncertainty, utility function is also utilized in our uncertainty tolerance engine, but we also include additional tools such as Bayesian probability (in the form of Bayes Net) and also Decision Networks in our solution.

2.2.1.1 Contingent pricing

Another mechanism to mitigate the issue of quality uncertainty is through the application of *contingency pricing*. Contingent pricing has its root in marketing and economics research area and is a common practice to reduce buyers' risk when dealing with future uncertainties related to purchase of goods and services such as travel related (air ticket, hotels, etc.), and entertainment (concert tickets, theme park admission) [27, 28]. Bazerman et. al. [25] discussed the benefits of contingent contracts in business negotiation to mitigate the issue of uncertainty in services.

A related paper is by Bhargava and Sundaresan [29] concerns on how to manage quality uncertainty through contingency pricing for IT related services. The issue in question has been long debated in economics, which is the difficulty to ascertain quality of services as compared to quality of goods. Bhargava and Sundaresan suggested the use of different pricing mechanism, termed as contingent pricing for different level of service quality offered to the customer. As noted by the authors, contingent pricing mechanism has been widely used in IT services. The authors contributions from the paper are the formal study of contingency pricing and the design of an optimal contingent contracts. Similar to Smith's paper above, Bhargava and Sundaresan put emphasize on solving the issue of quality uncertainty from the service provider perspective, whereby in our research, we look into the issue of uncertainty from the customer perspective. Secondly, their contingency pricing framework does not tolerate uncertainty prior to the service invocation, but mitigate the negative consequence of uncertainty after the uncertain event has occurred. Our approach of uncertainty tolerance in the service provisioning offer phase (in Chapter 5 addresses the issue of uncertainty prior to the service invocation.

2.2.1.2 Information Asymmetry

As mention earlier, the issue of uncertainty is not something new in other fields, and as for quality uncertainty, this has been discussed extensively for the past few decades in the field of economics. The economist, George Akerloff, in 1970 published a paper titled *"The Market for Lemons: Quality Uncertainty and the Market Mechanism* [30] which highlighted the issue of information asymmetry, which occurs when the seller knows more about a product being sold compared to the buyer. Although the core discussion about information asymmetry given in the paper relates to physical goods (automobiles), the same principle should applies to services as well (an example on health insurance, which is a form of service, is given as an applied example). In short, the information asymmetry is associated as quality uncertainty, whereby the seller has more information about the goods being sold, leading to two situations: (1) seller who knows that the goods has some negative quality, then try to sell the goods as a higher quality goods, or (2) seller who knows that their goods are of good quality but do not have the means to disclose this information to potential buyers. Akerlof's theory of *"market lemons"* has lead to *"lemon law"* (i.e. Magnuson-Moss Warranty Act [31]) which is a form of warranty law to tackle the issue of market lemons. Why this is important is that warranty [32] is also one type of solution against the issue of quality uncertainty. Apart from warranty, other mechanisms to combat quality uncertainty are moneyback guarantee [33, 34], and demonstrations [35].

On the issue of information asymmetry in Internet market, Pavlou et. al. in 2007 published a paper [36] that put the perspective of Akerlof *"market lemons"* theory into the online exchange (purchase/selling) relationship. The mechanism that they used to mitigate the issue of quality uncertainty resulting from information asymmetry is to use the principle-agent perspective [37, 38] based on a set of four uncertainty mitigating factors: (1) trust, (2) website informativeness, (3) product diagnoscity, and (4) social presence. Although the sources of uncertainty are identified in the paper, there is no proper classification scheme used to differentiate the type of uncertainty. Furthermore, although the scope of the problem has migrated from traditional physical market to the Internet or online market, the example given in the paper is still on the purchase of goods (books and drugs prescription) instead of pure services. Other related research on *"cyber lemons"* are by Huston and Spencer (coin auction through eBay) [39], and Pan (China's e-commerce market) [40, 41]. The issue of information asymmetry in grid computing is discussed in [42].

2.2.2 Trust and Uncertainty

The subject of *trust* is often related to uncertainty. Although the specific term *trust uncertainty* is not directly used in academic papers, there are several papers that discuss both the aspect of trust and uncertainty. For example, Viljanen [43] discusses thirteen different trust models for trust decision and one of the model, *confidence-aware* model has uncertainty associated with trust or the input factors. As for trust management, Ruohomaa and Kutnoven [44] publish a survey that discusses the general overview of the state of art in trust management. One of the statement in that paper which is related to this discussion is that trust plays and important role in virtual organization to counter the element of *uncertainty* due to business requirement for openness. Another key statement from Ruohomaa and Kutnoven is that trust is a way for people to deal with uncertainty when presented with decision making (future events) and when interacting with another party. Another paper related to trust management published by English et. al. [45], relates the uncertainty in trust values for decision making and trust evaluation. He suggested three alternatives for comparison of uncertainty of trust values: (1) ignore the uncertainty of trust values in both decision making and trust evaluation, (2) consider uncertainty of trust values only in decision making process, and (3) introduce the notion of uncertainty to the risk model, which caters for both the decision making and trust evaluation processes.

One specific paper which includes the keyword *"uncertainty"* and *trust* in the title is a paper by Brainov and Sandholm in 2000 [46] titled *"Contracting with uncertain level of trust"*. Although the area of research does not directly include the area of service computing, the scope of the problem is still applicable to our discussion since the paper addresses the problem of trust between two agents (in a multiagent system), a buyer and a seller, which is parallel to the concept of customer and service provider in service provisioning system. Unfortunately, although the keyword *"uncertain"* being used in the title, there is no definition of this keyword, nor there is indication or discussion of what is the relationship between uncertainty and trust. We assume that the authors refer to the inability to determine the trust level (trustworthiness) of their counterpart as the situation of uncertainty. Again, this paper exemplifies the needs for a definitive definition of uncertainty with respect to service provisioning.

2.3 Dempster-Shafer Analytical Hierarchy Process

Apart from Bayesian approach for solving subjective probability related problems, another approach which is usually discussed and compared to Bayesian approach is the

Dempster-Shafer Theory (DST). Additionally, and extension of the normal DST using Analytical Hierarchy process is also discussed in this section.

2.3.1 Dempster-Shafer Theory

The Dempster-Shafer Theory (DST) is a mathematical theory which falls under the subjective probability theory. The theory was developed by Arthur P. Dempster[47] in 1960s and extended and refined 10 years later by Glenn Shafer[48]. DST started to be used and popularized within the Artificial Intelligence (AI)[49] and Expert System fields as a technique for modelling reasoning under uncertainty. According to Beynon [50], real world practical applications of DST has been applied with success in areas such as face recognition [51, 52], target identification [53], intrusion detection [54, 55], and medical diagnosis [56].

DST allows the combination of evidence from different sources and generate a degree of belief (represented by a belief function) that takes into consideration all available evidence. DST framework allows the belief about a proposition to be represented between two values, *belief* (lower bound) and *plausibility* (upper bound). Under DST, beliefs from different sources can be combined using different fusion operators based on specific situations. There are several key differences between DST approach compared to Bayesian theory. First, evidence is represented as belief function rather than probability density function. Secondly, Bayesian Theory requires a more explicit formulation of conditioning and the prior probabilities of event. In DST, the conditioning information is embedded into the belief function and does not rely on prior knowledge. Third, the computation of evidence for a proposition does not required prior probability (i.e. assumes ignorance). According to Hoffman [57], there is no differences in term of performance between DST and Bayesian approach in managing uncertainty in sensing. Therefore, with regards to this project, there is no added advantage in using DST as compared to Bayes' Theory.

2.3.2 Analytical Hierarchy Process

Analytical Hierarchy Process (AHP) is a structured decision making framework for organizing and analyzing complex decisions. It was developed by Thomas L. Saaty [58, 59] in the 1970s and is based on the fields of mathematics and psychology. The AHP framework takes into consideration both the subjective and objective measurements in solving complex, multi-party criteria problems through the assessment of alternatives against an array of diverse objectives. According to Saaty [60], in a decision making process, the most creative part is in choosing the factors that are important for that decision. The main strength of AHP is that it allows a group of people (whom are involved in decision making process), to use their perception and judgment in the decision making process. The AHP framework is a popular tool in decision making problem especially in operations research [61, 62] and management science [63, 64].

2.3.3 Dempster-Shafer Analytical Hierarchy Process

Beynon et. al. in 2000 [50], introduced a method which combines both the aspects of the DST and AHP to solve multi criteria decision making problems. DST/AHP differs from AHP since it allows comparisons between group of decision alternatives instead of single alternatives, and includes DST's rule of combination. In essence, the DST/AHP approach allows measures of uncertainty and ignorance to be included as part of the decision making process of AHP. The DST/AHP approach has been applied in various situation such as sustainable transport solution [65], security risk assessment [66, 67], and urban power planning [68]. Although DST/AHP is useful in multi criteria decision making process, this approach is not relevant to this research since the evidence collected is of the same categories, and has only one parameter for decision making consideration.

2.4 Probabilistic Modeling of Networked Systems

Konnoy et. al. [69] defined a networked systems as a large and complex systems which require exhaustive work by the network administrators to maintain the quality and functions of the network system. Probabilistic modeling has been used to model, visualize, and understand the requirement, functions, and performance of network systems. For example, Guan et. al. [70] uses Bayesian Predictors (probabilistic model) on data collected by health monitoring tools when cloud servers perform normally, for proactive failure management of the cloud servers. Another research work that utilizes probabilistic modeling for performance evaluation is by Ghaffarkhah and Yasim [71], whereby a probabilistic model is used to determine channel characterization in mobile networks to ensure robust cooperative operation of the mobile network. Other related works are by Chemouil et. al. [72] (traffic routing in telephone network), Mao et. al [73] (link loss monitoring in wireless sensor networks), and Ni et. al. [74] (online risk-based security assessment).

Although our research works involve a network systems, in the form of service provisioning system, and utilizes a form of probabilistic modeling in the form of Bayes Theorem, we are more interested in addressing the uncertainty exhibited by the network systems rather than the aspect of performance of the network systems.

2.5 Summary

This chapter has reviewed several key research works which are directly related to the research work presented in this thesis. Two key findings that can be concluded from this chapter are: (1) there are very few researches that directly tackle the issue of uncertainty in service provisioning, and (2) there are researches that tackle uncertainty but in different form or perspective such as quality and trust. These findings provide sufficient motivation for the research works conducted in this thesis.

Chapter 3

Background

This chapter introduces two important concepts, (1) service and (2) uncertainty, which are the major focus of this thesis and provides the context for the problem of uncertainty in service provisioning. The concept of service provides the scope and context of the research problem whereby the uncertainty problem lies within the area of service, in particular service provisioning. One of the challenges is to distinguish between the concept of services and goods, which in turn affect the quality perspective of these two areas. One of the contributions of this chapter is the application of the Rathmell's Goods-Services Continuum test on a set of different type of electronic/web based services. As for uncertainty, this chapter explores the concept from the perspective of other research areas in which the issue of uncertainty is common. The insight from other research areas, especially on existing classification scheme will provide the foundation to design a unique classification scheme for the area of service provisioning. This hybrid classification scheme will be presented in the next chapter.

The remainder of this chapter is organized as follows: Section 3.2 will cover discussion on service and service provisioning concepts, including the definition of service and service provisioning, and the issue of service quality and Service Level Agreement. Section 3.7 discusses the concept of uncertainty in general and provides a thorough review of existing classification schemes in various research fields such as management, health care and so on.

3.1 The Area of Services Computing

The area of *Services Computing* has been established in November 2003 [75] and has become a cross-discipline research area that covers both science and technology in order to link the area of Business Services and IT services with the goal to perform services more efficiently and effectively. The area of services computing covers service-based computing such as web services, service-oriented architecture (SOA), cloud computing and also business aspect of such technologies such as business process modelling, transformation and integration. The 1st International Conference on Services Computing in 2004 with three major tracks: SOA and Web Services, Grid/Utility Computing, and e-Business Computing [76]. In the 1st Volume of the IEEE Transactions on Services Computing, Zhang [5] establishes 14 key body of knowledge in the area of services computing categorized into four categories: (C1) Services and services system, (C2) Services Technologies (C3) Services Consulting and Delivery, and (C4) Services Solutioning and Management. In general, we believe there is a gap within the body of knowledge of services computing that neglect the issue of uncertainty especially within a service lifecycle which is one of the knowledge area under the first category (C1).

3.2 What is a service?

Service has existed as an economic concept since time immemorial. Existing definitions of the term service can be viewed from two perspective, (1) economics and (2) technology. From an economic point of view Rathmell in 1966 [2] defined a service as the intangible equivalent of an economic product. In that paper, Rathmell distinguished between goods (owned or rented) and non-goods services using two tests. One distinction is to consider good as a *noun* and service as a *verb*, followed by several other distinctions as shown in Table 3.1.

The second test proposed by Rathmell is based on the nature of the product's utility. As for goods, the utility of the product (such as a book) is based solely on the good itself (no act is involved). On the other hand, the utility of a service (such as legal service) is solely based on the service rendered (no good is involved). One important observation by Rathmell is that in reality, both goods and services lies in a continuum, termed as *Goods-Services Continuum*, i.e. it is seldom that we can find pure goods or pure services. To add to this observation, Rathmell added that in reality, apart from

Table 3.1: Distinction between Goods and Services

Goods	**Services**
noun (good is a thing)	verb (service is an act)
an object, a device, an article, or a material	a deed, a performance, or an effort
when a good is purchased, a buyer acquire an *asset*	when a service is purchased, the buyer incurs an *expense*

these two extremes, both goods and services require each other in order to be useful. For example, most goods (food, books, etc.) require services (delivery, accounting, advertisement, etc.) to be able to be useful (i.e. goods being sold), and the same dependency applies to services as well.

As an exercise, we would like to apply Rathmell's *Goods-Services Continuum* test on a selection of electronic or web-based services. The objective of this exercise is to see whether the *Goods-Services Continuum* theory still applies to these type of services. The selected services have the following properties: (1) **web-based:** services delivered solely through the Internet or electronic medium, and (2) **paid services:** the services selected incur cost to the user/customer. Table 3.3 shows the compilation of type of service, example of service, operational objective, and the associated *Goods-Services Continuum* placement. The *placement* field refers to the possible placement of the type of service on the *Goods-Services Continuum.* Although there has been various service classification schemes proposed in the past [77], we classify the placement based on the operational objective of the service. Furthermore, we need to be clear what is defined as a *good.* The differentiation between goods and services has also been well researched by various researchers in early service marketing literature [2, 78, 79] and generally summarized as four characteristics (as shown in Table 3.2): (1) ***Tangibility:*** whether the product has physical presence, (2) ***Perishability:*** degree of durability beyond the time of purchase, (3) ***Separability:*** whether the product can be stored for later use, and (4) ***Standardization:*** whether quality can be controlled through standardization in the production process.

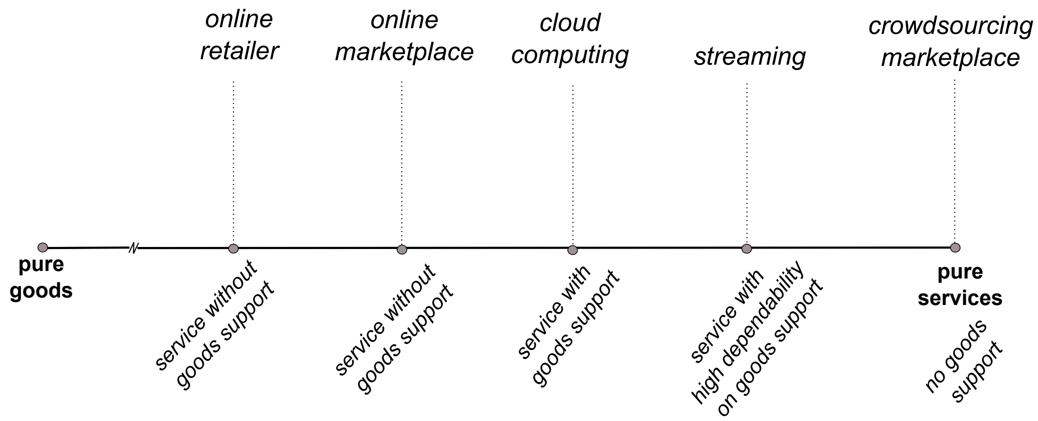


Figure 3.1: Placement of Selected Web-based Services on the Rathmell's *Goods-Services Continuum*

Firstly, *cloud computing* services such as Amazon Web Services (AWS)[1] and Rackspace[2] are defined as services which require goods support in order to be operational (leasing model), but no goods actually being sold. As for online retailer such as Amazon UK[3] and Play.com[4], they are considered as *services with high dependability* on goods support, whereby the service rendered by the retailers is to provide a virtual marketplace for the users and goods are actually being sold. For streaming (media) services like Hulu[5] and Netflix[6], these type of services can be considered as *services without goods support*, since no goods being sold. The other two types of services, *online marketplace* and *crowdsourcing marketplace* can be classified as *services without need for goods support.* For Amazon Mechanical Turk[7] service (crowdsourcing marketplace), this is a clear case of services without the needs of goods support since no actual goods being sold. If we look closely at online marketplace such as eBay UK[8], the service offered is for users to advertise and sell goods (there is an exchange of goods, i.e. goods being sold), but the question is, without goods, is it possible to offer the service (i.e. service cannot exists without goods support)? The placement of the selected web-based services is

[1]http://aws.amazon.com/what-is-aws/
[2]http://www.rackspace.co.uk/
[3]http://www.amazon.co.uk/
[4]http://www.play.com/
[5]http://www.hulu.com/
[6]https://signup.netflix.com/
[7]https://www.mturk.com/mturk/welcome
[8]http://www.ebay.co.uk/

Table 3.2: Differentiation between Goods and Services

	Goods	**Services**
Tangibility	are tangible (has shape and can be touched)	are non-tangible
Perishability	all goods have some degree of perishability	services in principle do not perish, only can be considered perished as they are consumed
Separability	goods can be stored for later use (production and consumption are separate)	in services, production and consumption occur at the same time
Standardization	quality of goods can be controlled during production	quality of services can vary each time they are delivered

Sources [2, 78, 79].

shown in Figure 3.1 (Note: location of placement does not indicates the quantitative strength of one placement to another).

The conclusion of this exercise is that we are able to differentiate between different type of web-based services using the Rathmell's *Goods-Services Continuum* test and also the four characteristics to make distinction between goods and services. Furthermore, based on the above discussion, another important aspect in the goods vs services discussion is the decision making process of selecting or accepting services as compared to to goods. This issue will be one of our focus in discussing uncertainty in service provisioning.

3.2.1 Definition of a Service

From technology perspective, there are several definitions such as *"electronic services"* (or e-services), and *"web services"*. As for the term e-service, several definition exists:

[9] https://www.mturk.com/

[10] http://www.smartsheet.com/ and http://en.wikipedia.org/wiki/Smartsheet

Table 3.3: Rathmell's [2] *Goods-Services Continuum* test on selected web-based services

Type of service	**Example**	**Operational Objective**	**Placement**
cloud computing	Amazon Web Services (AWS), Rackspace	provide users access to resources, software, and information over the Internet	service which require goods support (leasing model)
online retailer	Amazon, Play.com	to provide users access to goods (purchase) over the Internet	service with high dependability on goods support
streaming (media)	Hulu, Netflix	provide users with subscription service of movies, TV, and other type of media through the Internet.	service without goods support
online marketplace	eBay, Amazon Marketplace	provide a virtual environment (marketplace) for users to auction/sell goods	service without goods support
crowdsourcing marketplace	Amazon Mechanical Turk[9], Smartsheet[10]	to provide users access to human intelligence to perform tasks that a computer are unable to do yet.	service without goods support

Boyer et. al. [80] define e-service as interactive services delivered through the Internet using telecommunications, information, and multimedia technologies, while Rowley [81] classify e-service as any deeds, efforts, or performances conducted through the usage of information technology. A web service has been described as a software system that support interoperable machine-to-machine interaction over a network [82, 83]. In another definition by Kuebler [84], a web service aggregates one or more functionality for use. Zhang et. al. [85] define the term services as follow:

Definition 1 *"Services" represent a type of relationship-based interactions (activities) between at least one service provider and one service consumer to achieve a certain business goal or solution objective.*

We can see from the above discussion, there are multiple definition of the term *service*, either from different perspective or technology. Table 3.4 summarizes the various definitions of the term *service*.

3.3 Services Provisioning

In general, provisioning means providing or making something available. Historically, provisioning in information technology originates from telecommunication industry, whereby it is the process of preparing and equipping a network to allow new services to its users. In computing, provisioning appears in the context of utility computing, grid computing, service-oriented computing, and cloud computing.

3.3.1 Definition of Service Provisioning

Zhang's definition [85] views a service as a relationship between two entities but does not give any insight on the provisioning of the service itself. Therefore, we extend the view by providing the definition for *"service provisioning"*. In the context of the service provisioning through the Internet, we define service provisioning in general as follows:

Definition 2 *"Service provisioning" is the process of providing customers access to resources to complete tasks required by the customer. Resources can be in the form of hardware, software, or computation.*

Table 3.4: Summary of Definition of Service

Perspective	Term	Definition	Citation
economics		the intangible equivalent of an economic product	Rathmell, 1966 [2]
technology	e-service	(1) interactive services delivered through the Internet using telecommunications, information, and multimedia technologies	Boyer, 2002 [80]
		(2) any deeds, efforts, or performances conducted through the usage of information technology	Rowley, 2006 [81]
	web service	a software system that support interoperable machine-to-machine interaction over a network	Dustdar 2005 [82], Booth 2004 [83]
	service	represent a type of relationships-based interactions (activities) between at least one service provider and one service consumer to achieve a certain business goal or solution objective	Zhang, 2007 [85]

3.3.2 Motivation for service provisioning

There are different type of service provisioning such as hardware, application, and connectivity. For example, Software as a Service (SaaS) is a type of service provisioning that offer software or application to customer as a service on demand. On the other hand, utility computing provides end user with access to computation service to run computational intensive task such as financial modelling. The recent evolution of utility computing and the Internet is the emergence of cloud computing whereby shared resources, information, and software are provided to customer on demand. Examples of popular cloud computing implementation are Amazon Web Services (AWS) [86] and SalesForce [87].

Many organizations choose service provisioning because of the following benefits:

- **Reduced cost:** There is no need expensive investment to purchase hardware, software, and manpower for a one off task.
- **Ease of management:** Since the service provider provides management and technical support, the customer can concentrate on the business activity.
- **Rapid deployment:** Services is available instantly thus enabling of immediate deployment of the task.
- **Reliable services:** Since the service provider will provide a contractual agreement to guarantee the service provided.

3.3.3 Generic Service Provisioning Architecture

In this thesis, in order to simplify the scope of the research, a generic web service provisioning scenario as shown in Figure 3.2 is chosen as the basis for a service provisioning system. In a typical service provisioning architecture, there are three main entities [88, 89]: (1) service requester (client), (2) service provider (server), and (3) registry service.

In addition to these three entities, a fourth entity known as service broker (also known as facilitator and matchmaker) is sometimes present in certain system [90, 91]. The service broker acts as a middle agent which offers service from multiple service providers to a service requester. The added values of a service broker are as negotiation mediator [92] and synchronization [93]. Additionally, from the context of this research, the service broker entity can act as the neutral party that host the uncertainty tolerance engine and also can act as the evidence collection point for the uncertainty tolerance mechanism. This is assuming that the service broker is trusted and neutral (not bias) to both the service provider and service requester (consumer).

The three main entities function as follows:

- **Service Requester**: A service requester (generally known as customer or user) is an entity (which can be a human or software agent) which sought after a service to complete a task(s).
- **Service Provider**: A service provider is an entity that hosts and offers services to the end users (customer).
- **Service Directory**: A service directory or registry is essentially a service metadata portal for service registration and discovery.

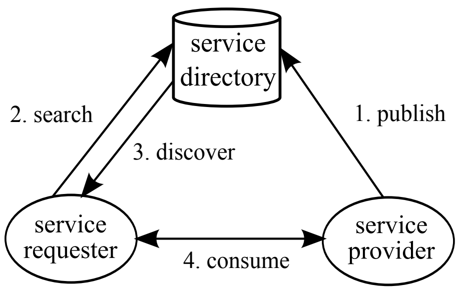


Figure 3.2: General Service Provisioning Architecture and Lifecycle

3.3.4 Service Provisioning Lifecycle

Figure 3.2 shows the basic architecture and life cycle of a service provisioning between the service provider and the end user. The first step (1) is for the service provider to publish the service into a registry. The registry acts like yellow pages that store information about services from various service providers. If a service is required, a customer can search (2) the registry for potential service. If a suitable service is discovered (3), the customer can then use (consume) the service. The life cycle completes when the service has been consumed (4), i.e. the required task is done.

3.3.4.1 Service Registration and Discovery

One of the challenges in the above service provisioning lifecycle is to discover services a customer wants among a collection of services and providers. The solution in the above lifecycle is to provide a registry service whereby a service provider can publish a service and this service can be discovered by an interested customer. One of the existing technology that provides service registry for web services is Universal Description Discovery and Integration (UDDI) standard. UDDI is a platform-independent registry standard that supports the publishing and discovery of web services. The UDDI standard is based on the Extensible Markup Language (XML). One of the elements of UDDI is the UDDI data structures for representation the provider and service description information. The UDDI XML Schema defines four types of information, which are:

1. ***businessEntity:*** a description of the organization that provides the service.
2. ***businessService:*** a list of all the Web services offered by the business entity.
3. ***bindingTemplate:*** describes the technical aspects of the service being offered.
4. ***tModel (technical model):*** is a generic element that can be used to store technical information on how to use the service, conditions for use, guarantees, etc.

The relationship between the different type of data structures is shown in Figure 3.3.

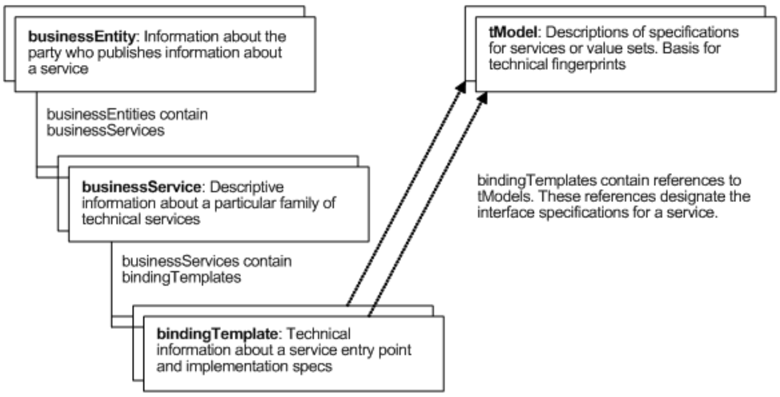


Figure 3.3: UDDI Data Structures (source [1])

The UDDI standard was first written in 2000 but unfortunately in 2006, due to poor adoption by the market, several important supporters of the standard such as Microsoft, SAP, and IBM decided to shut down their public UDDI nodes. The last surviving UDDI standard which is the UDDI v3 [1], was published in 2002 by OASIS (Organization for the Advancement of Structured Information Standards), the group that define, maintain, and sponsor the UDDI standard.

3.4 Service Quality and Service Level Agreement

One of the main challenges of service provisioning through the Internet is to ensure that the service offered meets the quality criteria specified by both parties. This a common issue in services as compared to defining and assessing quality of goods. Since goods has tangible properties, assessing goods quality is simply by judging the quality of the tangible properties (colour, label, feel, packaging, etc.). On the other hand, the unique characteristics of services contribute to the complexities involved in assessing and managing service quality. As a consequence, this issue complicate the consumers' assessment of quality and service provider's ability control and guarantee it. The issue of service quality has been discussed by various marketing researchers such as Groonroos [94, 95], Lewis and Booms [96], Parasuraman et. al. [97], and Mersha [98] which leads to two common agreements: (1) quality of service is much more complex and harder to determine compared to quality of goods, and (2) perception of service quality is the difference between customers' expectations and the actual service performance. Subsequently, another challenge is to measure the service quality itself. One of the well known methods for evaluating service quality has been developed by Parasuraman, Zeithaml, and Berry [99] known as SERVQUAL based on five quality dimensions (tangibles, reliability, responsiveness, assurance, empathy). The SERVQUAL method has also been tested on Information System (IS) service quality [100] and e-commerce businesses in tourism sector [101].

As for service guarantee for services-based computing system, one approach employed by service provider is to use a form of contractual agreement termed as Service Level Agreement (SLA). SLA is a common approach to provide service guarantee in various services-based system such as SLA for grid computing [102, 103], for cloud computing [104], for utility computing [105], and for web services [106]. Various researchers such as Kotsokalis [107], Ward et. al. [108], and Rana et. al. [109] have defined the term Service Level Agreement (SLA) and in general, their definitions consist of three important components: (1) it is a form of contract or agreement, (2) party involved: between a service requester (customer) and a service provider, and (3) agreement contains level of quality based on specific parameters. This leads to a generic definition of SLA as follows:

Table 3.5: List of Measurable Qualities

Quality	Brief Description
Accuracy	error rate of a service (average number of errors over a given specified time)
Availability	the mean time to failure for a given service (typically measured by the probability that the service is available when required)
Capacity	number of concurrent requests that can be handle by the service at any given time
Cost	the cost associated with each service request
Latency	the maximum amount of time between arrival of a request and the completion of a request.
Scalability	the ability of the service to increase the number of successful operations completed over a given time period.

Definition 3 *Service Level Agreement (SLA) is a contract (agreement) between the service provider and the consumer (end user) and define the service level required based on specific parameters.*

The quality specified in an SLA can be of any metric, provided it can be measured and verified by the parties involve in a service transaction. Bianco [110] categorizes quality into two groups: (1) Measurable qualities: can be measured automatically using a given metric (such as accuracy, availability, latency, etc), and (2) Unmeasurable qualities which cannot be measured from a given viewpoint in a service transaction (interoperability, security, etc). A non-exhaustive list of measurable qualities is shown in Table 3.5. In our research work, we are interested in the measurable qualities of an SLA.

In a competitive provisioning business environment, SLA can be a key commercial tool in attracting potential customer [111]. One of the key benefits of having an SLA is that it is an indication of the service provider confidence in its ability to deliver the service as promised. However, there must be a certain level of certainty that the SLA itself can be guaranteed and there is no uncertainties hidden from the SLA itself.

3.4.1 Elements and Structure of Service Level Agreement

The SLA document consists of several typical elements (as shown in Figure 3.4):

- parties (which include signatory and supporting parties).
- service description (service operations, SLA Parameters, metrics, measurement directives).
- obligations

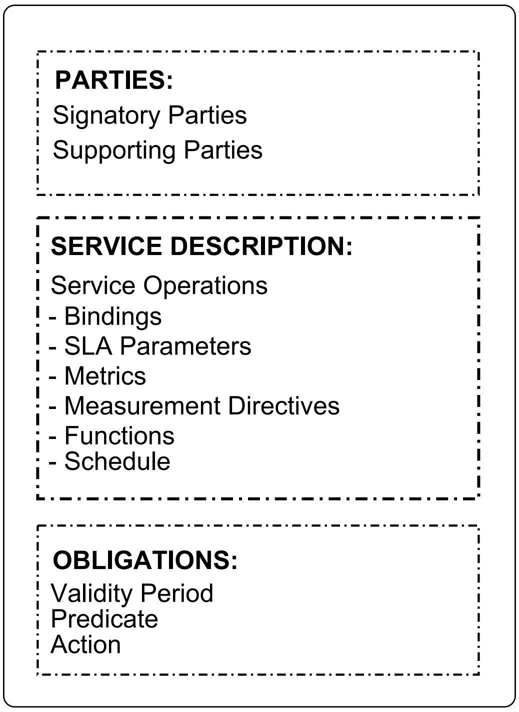


Figure 3.4: Typical SLA Structure and Elements

The *parties* section, which includes the signatory and supporting parties function as an identifier for all parties involved in the contract. The Signatory party description consists of the identification and the technical properties of the party (interface definition and addresses). The Supporting party description consists of the Signatory party description plus an attribute that represents the sponsor(s) of the party.

The *service description* section defines the characteristics of the service and its observable parameters. Each service operation contains one or more bindings (transport encoding for exchanged messages). Additionally, one or more SLA Parameters of the service can be specified here.

Definition 4 *An SLA Parameter is a defined property of a service object.*

Examples of SLA Parameters are “service response time”, “service throughput”, and “service availability”. SLA Parameters are comprised of composite Metrics (which is made of one or more other composite or resource metric).

- Examples of Composite Metrics: “maximum response time”, “average availability”, or ”minimum throughput”.
- Examples of Resource Metrics: “system uptime”, service outage period”, “number of service invocations”.

The last section, *obligation*, define various guarantees and constraints that can be enforced on the SLA Parameters. There are three main components in this section - (1) Validity period, (2) Predicate, and (3) Action. The first component (Validity period) specifies the time limit (duration) of which a given SLA Parameter is valid. The Predicate component indicates the threshold and the comparison operator (greater, equal, less, etc.) to be used to compare a computed SLA Parameter. The outcome of the Predicate can either be true or false. The final component, Action, will be triggered depending on the outcome of the Predicate. For example, if a violation of guarantee has occurred, (Predicate of the parameter is TRUE), then an action (for example opening a trouble ticket) will be triggered.

3.4.2 SLA Lifecycle

The life cycle of an SLA can be generally categorized into four phases - (1) creation, (2) deployment and provisioning, (3) enforcement, and monitoring of service invocation under an SLA, and finally (4) termination of SLA. However, depending on the business scenario, there may be many sub phases within each phase. Figure 3.5 below shows a more detailed lifecycle of an SLA lifecycle with sub-phases.

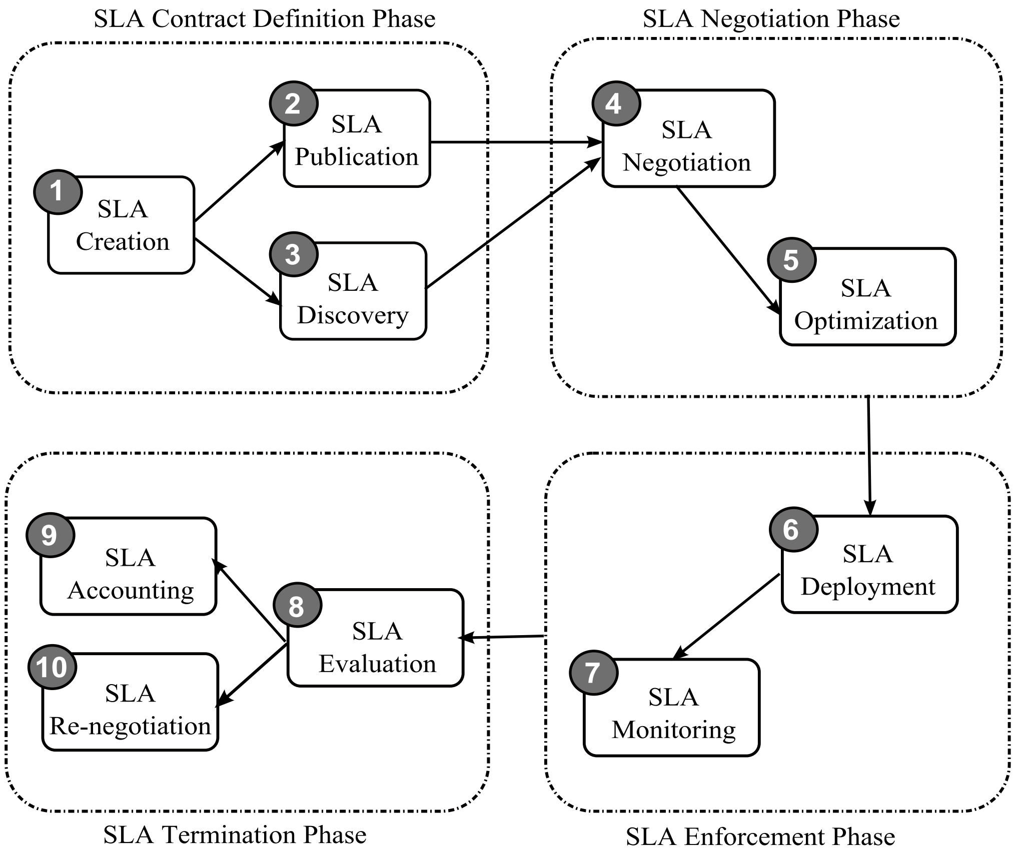


Figure 3.5: SLA lifecycle with sub-phases

3.5 SLA Frameworks and Languages

Organizations refer to Service Level Agreement Frameworks or SLA Frameworks as a phrase that refers both to the scope of services to be covered by the SLA plus the monitoring and governance process that are put in place to ensure the compliance of the SLA. There are two main existing frameworks for SLA specification and monitoring: (1) Web Service Level Agreement (WSLA) developed by IBM, and (2) Web Service Agreement (WS-Agreement) developed by a working group of the Open Grid Forum (OGF).

3.5.1 Web Service Level Agreement (WSLA)

Web Service Level Agreement (WSLA) is a standard for service level agreement compliance monitoring of web services and version 1.0 was published by IBM on January

2001 [18, 112, 113]. It allows the service provider to specify performance metrics associated with a web service application and also how the metrics are measured. The framework is capable of measuring and monitoring the QoS parameters of a web service, and reports any violations to the entities or parties specified in the SLA. The SLAs are expressed using a formal language which is based on an XML Schema, and is interpreted by a runtime architecture in the framework. This runtime time architecture consists of several monitoring modules, which can be located in an external party such as a trusted third party.

The are several projects, and researches that utilizes WSLA as SLA framework or as the basis for improvement to cater for specific domain or needs. For example, the GEMSS [114] project embedded the WSLA framework in its grid service that supports the provision of medical simulation services by service providers to clients such as hospitals. Another research projects that utilizes the WSLA framework for commercial grid environment is proposed by Leff and Rayfield [115].

WSLA works well to cater for scenario that involves two parties with distinct role as service provider and consumer, but does not provides support for dynamic collaboration environment whereby there are multiple parties with similar/different roles, and multiple services. Therefore, Nepal et. al. [116] proposed WSLA+ which is an extension to WSLA to support SLA requirements in multi-party collaborations.

3.5.2 Web Service Agreement (WS-Agreement)

The WS-Agreement (Web Service Agreement) specification [117] published by the Open Grid Forum (OGF) focuses on grid computing environment. The main objective of WS-Agreement specification is to provide a contract or agreement between two parties, such as service provider and consumer, using an extensible XML language for specifying the nature of the agreement. An agreement between the service provider and the consumer defines the relationship between the two parties that is dynamically established and managed. In the agreement, each party agrees on the roles, rights, and obligations. From the provider point of view, a provider in an agreement provisions the service following the conditions described in the agreement, while the consumer enters into the agreement with the aim of obtaining guarantees on the availability of the service provided by the service provider.

The specification also provides templates to enable the discovery of compatible agreement parties. There are three main parts of the specification: (1) a schema for specifying an agreement, (2) a schema for specifying an agreement template, and (3)a set of port types and operations for managing agreement life-cycle, and monitoring of agreement states. The WS-Agreement specification has been successfully implemented in the Globus Toolkit 4.0 environment [118]. Additionally, WS-Agreement provides negotiation capability which can be used by either the service provider or the consumer, or by another third party on behalf of service provider and/or consumer.

3.5.3 Other Frameworks

Other SLA Frameworks and languages exist but are limited as research project, or as an extension from either WSLA and WS-Agreement frameworks. Furthermore, many service providers have started to offer services or migrate existing services to cloud environment from traditional IT infrastructure. Due to the dynamic nature of cloud environment, and other factors such as trust, the existing SLA frameworks is not sufficient to provide SLA specification and monitoring. Thus, there are several researches that look into the needs for an SLA Framework which is specific to cloud computing environment. Patel et. al. [119] proposed a mechanism for managing SLAs in a cloud computing environment utilizing the existing WSLA framework. To solve the issue of trust in a cloud environment, the monitoring and enforcement of the SLA is delegated to a trusted third party using the existing feature in the WSLA framework.

Web Service Offerings Language (WSOL) [120] is another SLA framework that provides formal specification for different classes of services and constraints for web services. The author claims that the specification is compatible with existing web services standards and adds support for management of web services, their compositions, and selection of web services with different classes of service. WSOL is geared towards mobile and embedded web services [120] which have specific requirements such as limited run-time memory, limited power supply, and slow wireless links. Additionally, WSOL can handle context-sensitive situations (geographic location, time zone differences, etc.) and frequent disconnections.

Lamanna et. al. [121] from UCL has proposed and developed SLang which is a language to define Service Level Agreement that accommodate end-to-end quality of service. Similar to WSLA and WS-Agreement, SLang uses XML to define SLAs.

According to Wu and Buyya [105], there are three main differences between SLang and WSLA. Firstly, SLang contains an SLA vocabulary for defining Internet services. Secondly, SLang complements existing e-business industry standards such as ebXML [122] and BPEL [123]. Finally it is modelled using Unified Markup Language (UML) and definition is based on behaviour of services and consumers. Additionally, SLang provides the notion of vertical and horizontal SLAs which is a classification of interaction between the entities involved in the services. Vertical SLAs govern the interaction between subordinated pairs while horizontal SLAs govern the interaction between coordinated peers.

From the above discussion, the it is clear that the issue of uncertainty is not catered within any of the existing SLA frameworks and languages. Therefore, this project will provide a valuable insight into the issue of uncertainty in service provisioning and the possibility of adding uncertainty tolerance mechanism in the existing SLA framework.

3.6 Service SLA Monitoring and Violation

Both monitoring and violation activities are important to ensure the compliance of SLA which is agreed between service provider and service requester. For service SLA monitoring, understanding the challenges related to monitoring is important since some of the challenges are linked to the uncertainty tolerance approach which will be discussed in details in Chapter 5. As for service SLA violation, the activity itself does not give rise to the issue of uncertainty, but understanding the process is important since it is part of validating service compliance.

3.6.1 Service SLA Monitoring

Given an SLA for a service, there is a need to monitor the service to ensure that the agreed SLA between service requester and service provider is met or complied. There are various challenges in SLA monitoring such as: (C1) suitable location for monitoring module, (C2) accurate measurement of SLA, (C3) efficient monitoring and measurement of SLA, (C4) truthful reporting of the SLA monitoring. For the purpose of this thesis, we are only concerned about challenges C1 and C4.

The first challenge, the suitability of the location for the monitoring module concerns the issue of trust and also related to challenges C2 and C3. The issue of trust arises since the placement of the monitoring module within either service provider or service requester does not provide mutual trust to both parties. The different possible locations for the monitoring module is distinguished by Rana et. al. [109] as follows:

- **Trusted Third Party:** the TTP is an independent entity that can monitor and log activities between the service requester and service provider location. The key requirement is that the TTP has to be trusted by both *SR* and *SP*. To provide non-repudiation and reputation for both *SR* and *SP*, a signed ticket is generated after the service has been completed and send to both parties. One drawback of TTP location is that it is not possible to monitor activities internal to either the *SP* or *SR*. Keller et. al. [113] discussed the issue of delegating the role of service monitoring to third party entity. This is required when neither the service provider nor the service requester can be trusted to perform the monitoring role or wants to perform the role.

  The actual location of the service monitoring is not specified in the generic architecture discussed in Figure 3.2, Section 3.3.3. One potential solution is to combine the service directory entity with the monitoring service in a single physical location. One drawback is that the monitoring service can create a bottleneck if the monitoring service is physically colocated with the registry service. In real world scenario, the UDDI (Universal Description Discovery and Integration) standard (discussed in Section 3.3.4.1) provides the registry service and the monitoring service can be implemented separately using other technology/approach such as using WSLA framework [113] or SALMon [124] framework, but within the same physical location.

- **Service Provider:** The second option is to implement the monitoring module at service provider location which has equivalent functionality as TTP, i.e. able to monitor and log activities between itself and service requester, but with the advantage of monitoring the internal activities or state of the service provider. However, there are two drawbacks: (1) *SP* might not revealed or report the full information about its internal state and selectively choose information which is beneficial to itself, and (2) *SP* might falsely report the actual outcome of the monitoring (i.e. whether violation occurs for SLOs).

- **Service Requester:** The third option is to implement the monitoring module at the service requester location. In term of functionality, it is equivalent to the TTP, in which the service requester needs to determine if a SLO[1] violates the agreement. Unfortunately, to prove such violation to the TTP and SP is difficult. For example, in monitoring delay SLO, the service requester is unable to distinguish between network delay (which is not under *SP* controls) and processing delay in *SP* location. Therefore, the value for implementation of monitoring module at service requester location is to provide a means to establish a measure of trust towards the service provider.

The above discussion on the suitability of location will be used to justify the placement of the uncertainty tolerance mechanism which will be discussed in Chapter 6. The consensus is that the TTP is the most suitable location for the monitoring module if trust is the key characteristic [106]. There is also an attempt to combine both the advantages of both the client (service requester) and server side (service provider) monitoring as suggested by Michalmayr et. al. [125].

3.6.2 Service SLA Violation

An SLA contains one or more quality of service metric, termed as Service Level Objective that must be fulfilled by a service provider. In order for these SLOs to be fulfilled, they need to be *measurable* and *monitored* (discussed in previous section) during the provision of the service. In short, the term *violation* refers to the failure to fulfil or achieve the agreed level for SLO which in turn affect the overall SLA compliance. Rana et. al. [109] distinguished three type of provisioning in relation to SLA violation: (1) *'All-or-nothing' Provisioning:* where all SLOs in an SLA must be fulfilled in order for the SLA to be compliant with, (2) *Partial Provisioning:* where some SLOs are mandatory and must be fulfilled in order for the SLA to be compliant with, and (3) *Weighted Partial Provisioning:* where an SLO is met if it exceeds the threshold defined by the client. In relation to this thesis, the *'All-or-nothing'* approach is employed in Chapter 5 and Chapter 6 to detect violation and validate SLA compliance.

Another aspect of SLA violation is the inclusion of *penalty* which is a form of financial compensation from the service provider to a customer in the event of violation.

[1]Service Level Objective

Various aspects of penalty such as type of penalty, how to include penalty clause in the SLA, the amount of penalty in term of monetary or other form of financial compensation (credit, etc.) in relation to the violation, and impact to negotiation/renegotiation of SLA is addressed by Rana et. al in [109]. In the context of this thesis, we do not take into account the affect of penalty towards the issue of uncertainty.

3.7 Uncertainty

In our daily life, we always face situations that are not completely predictable. If I want to drive to the city centre tomorrow and I do not want to be late, I cannot be certain that there will be congestion or traffic problem that will delay my journey. I can contact relevant authority and get forecast of the traffic, which might be based on data collected over a period of time, or maybe any specific event (for example a football match) that can occur tomorrow. Based on the forecast (which indicate all is well), I might decide to get out from the house as usual. Despite the forecast, traffic delay can still occurs and I will be late for my appointment. Therefore, there is always uncertainty, but the question is can we tolerate them?

This section will provide a general discussion on three general aspect of uncertainty, which are: (1) definition of uncertainty, (2) sources of uncertainty, and (3) classification of uncertainty. One of the main challenge is that the three aspects varied greatly across different and similar research areas. Our objective is to gain an insight on how each research area defines these three aspects in order to provide our own version (of definition, sources, and classification) with respect to service provisioning.

3.7.1 Definition of Uncertainty

The term ***uncertainty***, in general refers to the condition of being unsure about someone or something [126]. Although this term is widely used by the general public, there are different definitions in different specialized fields such as physics, economics, sociology, engineering, and information science. Definition from other fields is as follow:

- **Decision Making:** Situation where the current state of knowledge is such that (1) the order or nature of things is unknown, (2) the consequence, extent, or magnitude of circumstances, conditions, or events is unpredictable, and (3) credible probabilities to possible outcomes cannot be assigned [127].

- **Information theory:** Degree to which available choices or the outcomes of possible alternatives are free from constraints [128].
- **Statistics:** Situation where neither the probability distribution of a variable nor its mode of occurrence is known.
- **Hard sciences (physics, chemistry, etc) and Engineering:** the interval of confidence around the measured value such that the measured value is certain not to lie outside this stated interval [129].
- **Economics:** uncertainty refers to the risk that is immeasurable, not possible to calculate [130].

From the above various definitions, it seems that there is no consensus on the precise definition of uncertainty. This is no surprising, due to the nature of uncertainty which can be subjective in nature. We will reserve our definition of uncertainty for service provisioning in the next chapter, since it is important to provide an abstraction of the service provisioning system beforehand.

3.7.2 Sources of Uncertainty

Uncertainty may arises from different sources. Knowing and understanding the sources of uncertainty is important since it would help to select the best method to treat or tolerate the uncertainty. In general, there are five sources of uncertainty (as shown in Figure 3.6): (1) Incomplete information, (2) Statistical variation, (3)Randomness and variability, (4) Linguistic imprecision, and (5) Frame of reference. Some of the following sources are discussed in details in [131].

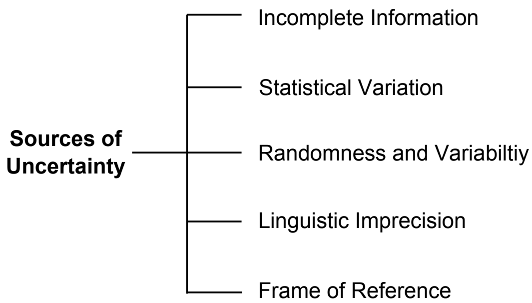


Figure 3.6: Sources of Uncertainty

Incomplete information: Incomplete information or knowledge can give rise to uncertainty when a factor in a decision or model is simply not known at that point of time. In certain cases of incomplete information, these can be resolved by research, inquiry, etc. For example, if the number of population of UK is a factor in a certain model, this information can be obtained. On the other hand, some factors are indeterminate (future events or developments, eg. number of death due to lung cancer in 2045) or practically immeasurable (eg. number of people smoking at the moment).

Statistical variation: This source of uncertainty comes from the direct measurements of a quantity, due to the physical, and technical limitation of measuring instruments, techniques, tools, and software. For example, when recording a delay in a service response, smaller temporal offset can be introduced by system processing (data packet has to travel through network interface card and probe application before being recorded) [132].

Randomness or variability: There are factors or quantities which have inherent randomness that give rise to uncertainty. For example, in Heisenberg Principle of Uncertainty [133], the position and velocity of an electron cannot be known in advance. Other quantities, although not naturally random, has to be treated as such due to our inability to compute or measure them accurately enough. For example, weather prediction is affected by multiple initial conditions [134].

Linguistic imprecision: Human language (written or spoken) can be a source of uncertainty if imprecise terms and expressions is used. Furthermore, language is also affected by various factors such as contextual, and cultural. For example, if phrases like *"highly likely"* or *"high probability"* are used without specific values, can lead to uncertainty.

Frame of reference: given a situation or choice, two different persons can have different belief due to the fact that they have different "frame of reference". For each person, the "frame of reference" is based on multiple factors such as facts (information and knowledge), interests, norms, and values that are gathered through previous experiences and perceptions. Koppenjan [135] discusses this further in relation to analysis of uncertainties in dealing with complex problems.

3.7.3 Classification of Uncertainty

There have been many attempts at classifying different types of uncertainty by researchers for various purposes, unfortunately there seem to be no agreed classification that can be used by all. This is expected since there are various definitions of uncertainty as discussed previously. Hence, we notice that existing classification schemes are very much dependent on the area or field of research. Some classification schemes (as discussed below) cover broad area of research, and others are very specific, for example for water resources research [136]. We also notice that there are some overlaps in the classification scheme being used in different fields. Our aim for this section is to get an insight on the classification scheme of uncertainty from various fields, to enable us to device our own classification with respect to uncertainty in service provisioning. The ability to classify uncertainty into different type is important since this will lead into probable methods of solving the specific uncertainty being investigated. Furthermore, a singular classification scheme (for uncertainty in service provisioning) will enable researchers to have the same understanding on the problem. We will discuss some of the more widely used classification scheme as follows.

3.7.3.1 In engineering, design process, and risk assessment

In engineering related fields(risk assessment[137], reliability engineering [138, 139]) classification is divided into two broad extremes, which are (1) epistemic, and (2) aleatory uncertainty. Aleatory uncertainty (AU) is an inherent variation associated with the physical system or the environment. It is also can be referred as variability, irreducible uncertainty, stochastic uncertainty, or random uncertainty. On the other hand, Epistemic Uncertainty (EU) is an uncertainty that is due to the lack of knowledge of quantities or processes of the system or environment. It is also known as subjective uncertainty, reducible uncertainty, or model form uncertainty. Examples of epistemic uncertainty are lack of experimental data and poor understanding of initiating events.

As we will see in subsequent sections, every field seems to have a variation or totally different classification scheme. For example, in risk assessment (of radioactive waste repositories), Zio and Apostolakis [140] proposed a classification for model uncertainty that can be classified into three types: (1) conceptual model, (2) mathematical model, and (3) computer code (computer model). Although the classification scheme seems different (from epistemic/aleatory classification), it is actually a subset of epistemic uncertainty as clarified by Zio and Apostolakis. In essence, the model being classified is an *"epistemic probability model"* that represents knowledge about parameter values and model assumptions.

Clarkson [141] groups uncertainty into two basic types in relation to (engineering) design process research area: (1) known uncertainty, and (2) unknown uncertainty. He describes *known uncertainty* as the variability in past cases which can be characterized by probability distributions and *unknown probability* is related to "unmeasurable" differences in measurement. Furthermore, these two types of uncertainty are present in two areas: (1) description, and (2) data. Uncertainty in description encompasses selection of element, naming, ambiguity of description, and uncertainty in data lies in accuracy, completeness, and consistency in the design process. Clarkson's classification scheme is illustrated in Figure 3.7.

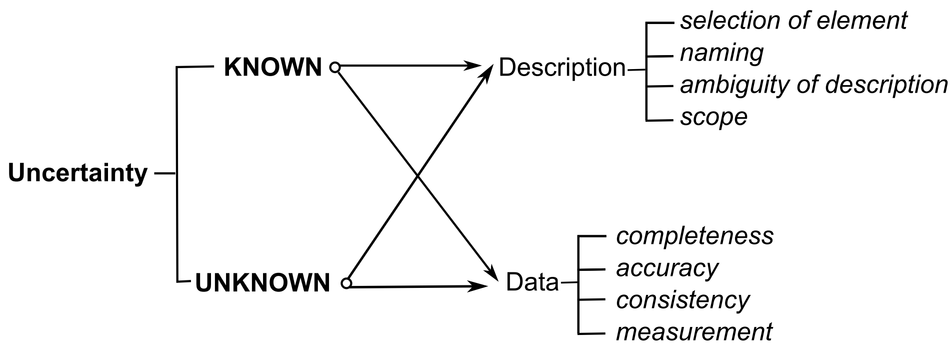


Figure 3.7: Clarkson's classification of uncertainty in engineering design process

There is also an attempt within the (engineering) design process research fields to create a somewhat generic (or holistic) classification scheme as proposed by Kreye et. al. [142]. The aim of the paper is to provide a classification scheme for the manifestation of uncertainty in design process, whereby the authors define *manifestation* as *"the point of the process where the uncertainty occurs"*. The manifestation uncertainty is one (out of five layers, the other four are nature, cause, level, and expression) of the layer in a *holistic* classification scheme, introduced by the Kreye et. al. and briefly discussed in the same paper. Furthermore, the term *manifestation* is similar to the term *location* in Walker et. al. [143] classification scheme (in Section 3.7.3.3). The authors created the holistic classification scheme by reviewing various other classification schemes across different domains (design, metrology, economics, and management). As for the manifestation uncertainty, the classification has four *points of occurrence* with sub-classification of each category. The four *points of occurrence* are: (1) context uncertainty, (2) data uncertainty, (3) model uncertainty, and (4) phenomenological uncertainty. Please refer to Figure 3.8 for overall view and subcategories of the classification. To illustrate the application of the (manifestation uncertainty) classification scheme, the authors applied the classification to thirty four papers in ICED[1] proceedings from 2003 to 2009. For further discussion on the classification and subcategories, please refer to Kreye et. al. paper.

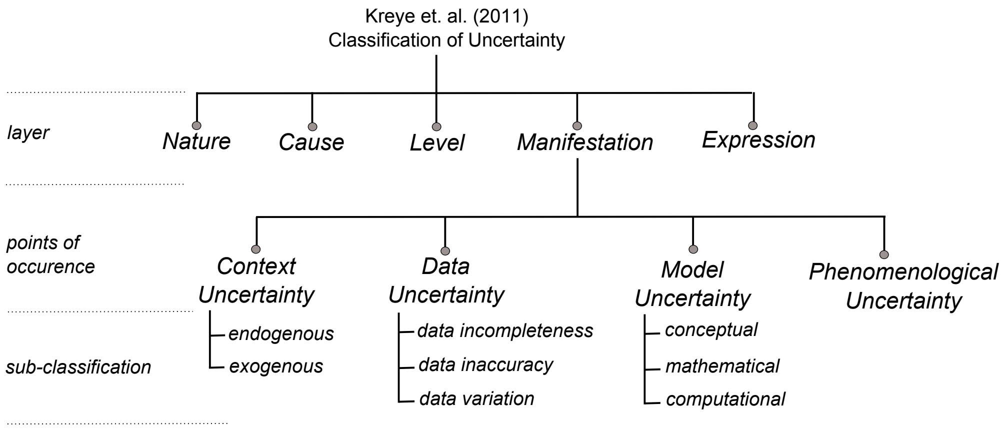


Figure 3.8: Kreye et. al.'s classification of uncertainty in design process

3.7.3.2 In cognitive science and psychology

Another classification scheme is based on (1) internal, and (2) external types, which is used in the field of cognitive science and psychology. For example, Howell and Burnett [144] describe *internal uncertainty* as an event whereby a subject has some control, and *external uncertainty* where a subject has no control. Kahneman and Tversky [145] also uses the internal and external classification, but describe *internal uncertainty* as a subject's state of knowledge and *external uncertainty* which is attributed to the real (external) world. They further classify *internal uncertainty* into two modes, (1) distributional mode (instance of a class of similar cases), and (2) singular mode (probabilities

[1]International Conference on Engineering Design

assessed by propensities of a particular case). Similarly, the *internal uncertainty* is classified into two modes, (1) Introspective, and (2) Reasoned. Kahneman and Tversky's classification is illustrated in Figure 3.9.

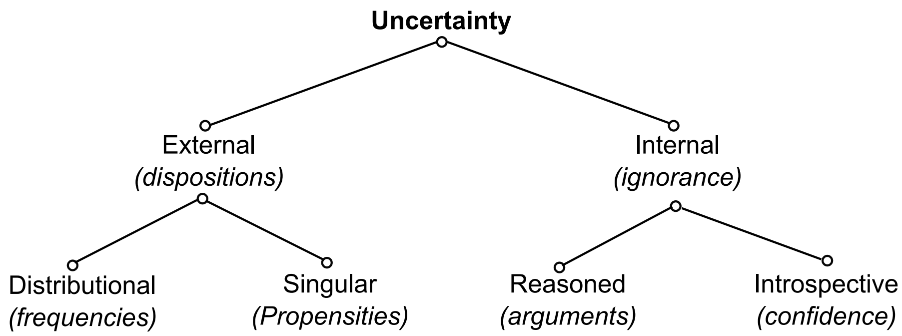


Figure 3.9: Kahneman and Tversky's Classification of Uncertainty

Several researchers [146, 147, 148] use the term exogenous and endogenous, which respectively refer to external and internal uncertainty. Weck [146] discussed exogenous and endogenous uncertainty from the product and system design perspective and add an important distinction to the classification in term of system boundary. Howell & Burnett, and Kahneman & Tversky implicitly define the boundary for the internal and external classification as the subject itself, while Weck defines the boundary as *system boundary* or *sphere of influence*. This definition enable the researcher to define a *system* or a *sphere of influence* as a group of different factors or components.

3.7.3.3 In decision support system

Another interesting classification scheme has been proposed by Walker et. al. [143] for classifying uncertainties in models used for decision support system. The classification scheme is based on three dimensions: (1) location (where the uncertainty exists in the model complex), (2) nature (where uncertainty classified as epistemic or variability), and (3) level (where uncertainty exists as as a range between determinism and indeterminacy). Furthermore, each of these dimensions is further divided into classes and sub-classes. Walker et. al.'s classification scheme is illustrated in Figure 3.10. From Figure 3.10, we can see that the *nature* dimension which is divided into two classes, (1) epistemic, and (2) variability is similar to classification from engineering related field as discussed above. We conclude that Walker et. al. classification scheme is very comprehensive but some of the sub-classification is probably too specific, catered for policy making and decision support system. for example, the sub-class *scenario uncertainty* in the *level* dimension is specific for policy making process. Further details and discussion of this scheme (discussion on the classes and sub-classes) is available from [143] and further discussion from [149]. Walker et. al. also suggest that uncertainty is a three dimensional concept based on the three dimensions discussed above, which results in the creation of an *uncertainty matrix* which can be used as a tool to get a graphical overview of decision support activities.

3.7.3.4 In health care related fields

In health care related fields there are several different classification scheme. For example, in nursing, Mishel [150, 151] has developed a classification for perceived uncertainty in illness, based on four dimensions (refer to Figure 3.11): (1) *ambiguity* - as patient's self-evaluation of uncertainty in illness as vague or unclear, (2) *complexity* - different information that patients receive about treatment and care system, (3) *deficient information* - inadequate information about patients' diagnosis, and (4) *unpredictability* - variability in patients' illness and outcomes. These four dimensions closely linked to an individual's (patient) state of mind. Although Mishel's classification is useful in identifying factors that contribute to uncertainty in health care, the classification lacks detail definition of the factors. Another classification proposed by Babrow et. al. [152], based on five dimensions of meaning of uncertainty (refer to Figure 3.11): (1) *complexity* - due to multicausality, contingency, reciprocity, and unpredictability, (2) *qualities of information* - due to clarity, accuracy, completeness, volume, ambiguity, consistency, applicability, confidence in sources, (3) *probability* - due to belief in specific or range of probabilities, (4) *structure of information* - due to order, integration, and (5) lay epistemology (individual's belief about a phenomenon). Babrow's classification is more expansive compared to Mishel (though the dimensions used are different) since each dimension has a defined contributing factors. In another example of uncertainty classification, Kasper et. al. [153] developed a conceptual classification to address decision-related uncertainty related to cancer patients. The study conducted by Kasper et. al. yield a classification of uncertainty with eight categories (refer to Figure 3.11): (1) social integration, (2) diagnosis & prognosis, (3) deciphering information, (4) mastering of requirements, (5) causal attribution, (6) own preferred degree of

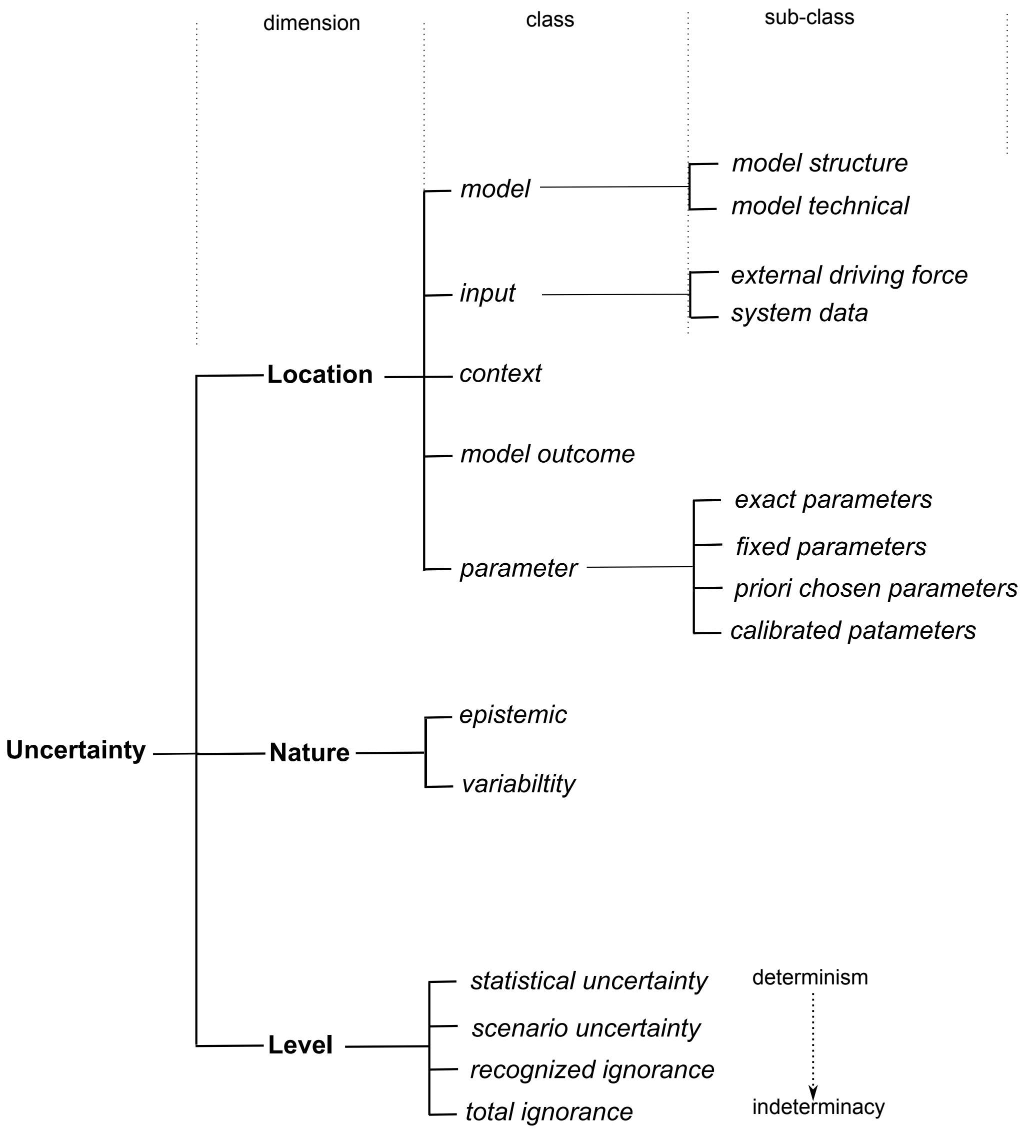


Figure 3.10: Walker et. al.'s classification of uncertainty in decision support system

involvement, (7) physician's trustability, and (8) treatment. Please refer to respective paper by Mishel [150, 151], Babrow et. al. [152], and Kasper et. al. [153] for detail discussion, and examples for the classification schemes. These classification schemes very much rely on cognitive state and also include specific dimension (or factors) related to clinical problem (such as diagnosis, prognosis, treatment, etc.).

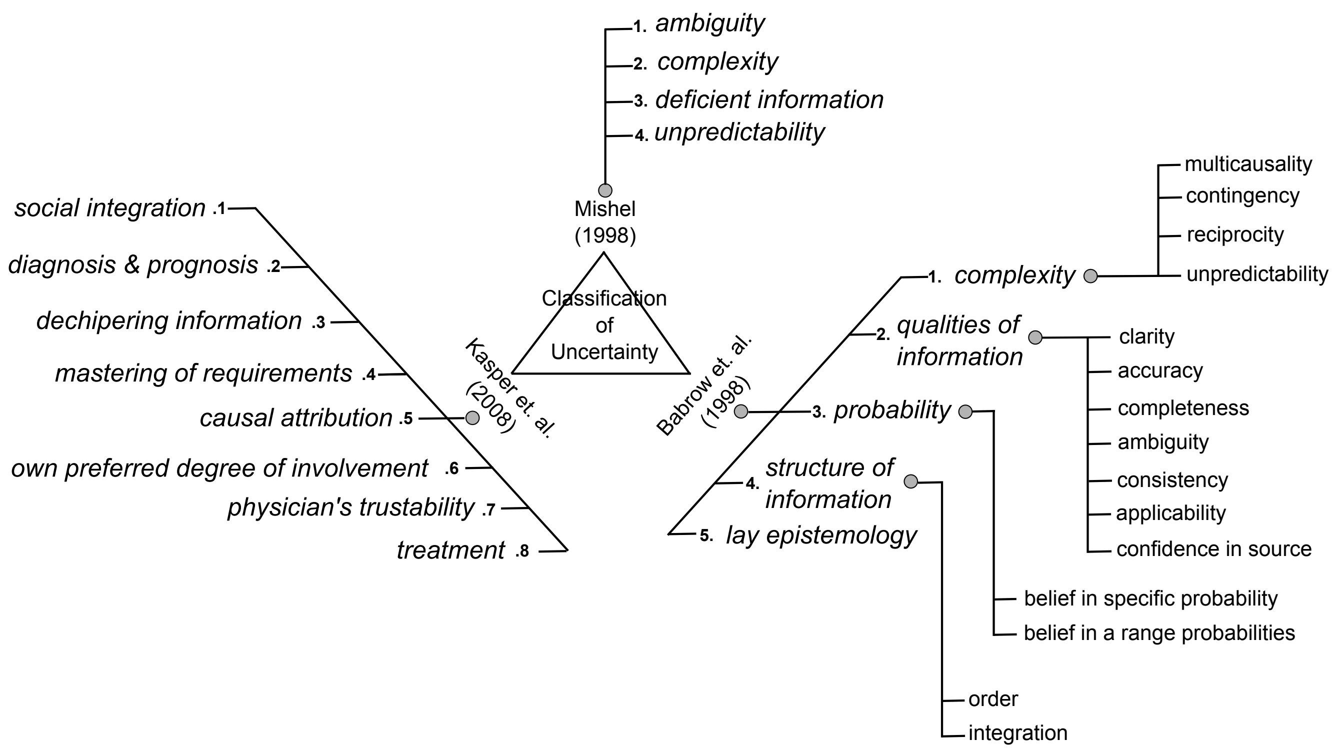


Figure 3.11: Classification of uncertainty in health care related fields.

3.7.3.5 In management related fields

Another research area that does have a variety of classification is management. Priem et. al. [154] provide a brief review on previous efforts to classify uncertainties related to organizational environment. Rather than repeating the review, one classification from the paper is taken to give an insight on the classification scheme. Two things to note from Priem et. al.'s paper: (1) the paper main aim is to develop *"numerical"* taxonomy of uncertainty sources from empirical data on executive perceptions, and (2) the term *environment* is used to refer to class or dimension for the purpose of grouping.

3.7.3.6 Hybrid/generic Classification scheme

There is also an attempt to create a classification of uncertainty that can be applied to all field of research, as proposed by Lo and Mueller [155]. Although their aim is to provide a taxonomy to cater for economics and finance, Lo and Mueller claim that their taxonomy covers uncertainty across different academic fields such as physics, biology, economics, philosophy, and religion. This taxonomy has five levels: (1) Level 1: complete certainty, (2) Level 2: risk without uncertainty, (3) Level 3: fully reducible uncertainty, (4) Level 4: partially reducible uncertainty, and (5) Level 5: irreducible uncertainty.

For Level 1, Lo and Mueller suggest that the field of classical physics fit the picture of complete certainty. For example, using Newton's laws of motion, if the initial conditions are fixed and known, then all past and future states can be exactly determined, i.e. *complete certainty*. Next, in Level 2, the randomness of uncertainty is perceived as similar to Knightian Uncertainty (Frank Knight, 1921)[130] where the randomness is dictated by a known probability distribution for a complete known set of sample space (outcomes). Furthermore, Lo and Mueller argue that since the probability distribution, rules, and odds are all known, there is no need for statistical inference. This classification level is useful for analysing risk. Level 3 is associated with risk that has a degree of uncertainty, due to unknown probabilities in completely known outcomes. Unlike Level 2, in order to conduct analysis, classical (frequentist) statistical inference need to be used together with probability theory. As for Level 4, Lo and Mueller claim that there is a limit to which uncertainty can be deduce for certain situations (for example data-generating process) that exhibit certain characteristics (refer to Figure 3.12 for list

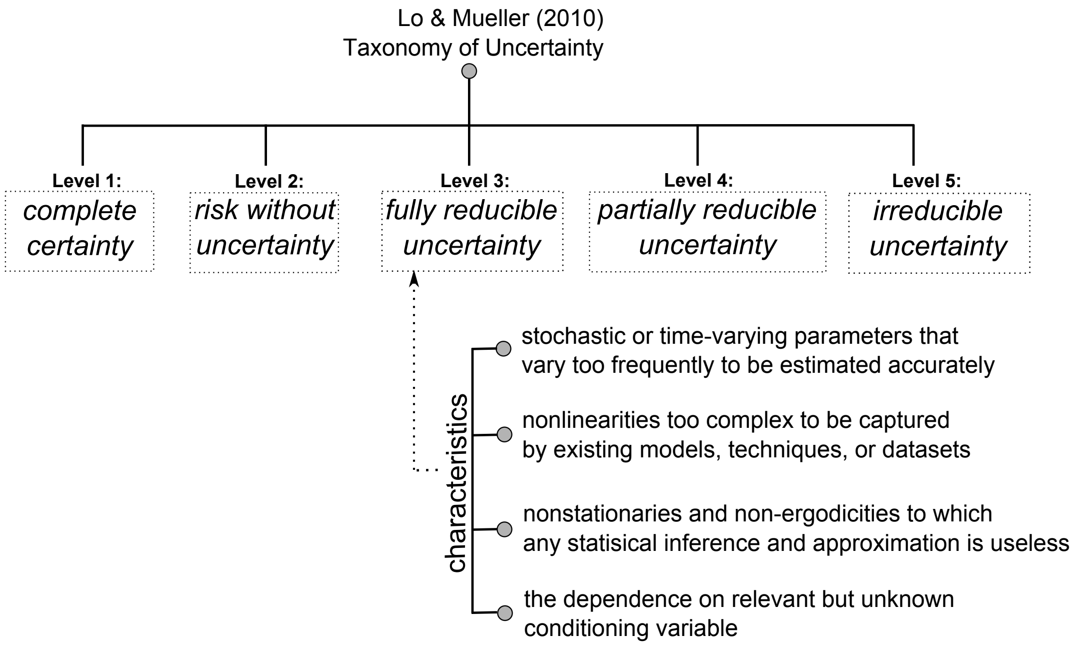


Figure 3.12: Lo & Mueller taxonomy of uncertainty

of characteristics). Finally in Level 5, irreducible uncertainty is liken to total ignorance whereby it cannot be solved by collecting more data nor using sophisticated methods of statistical inference (or any any other means). Lo and Mueller claim that this type of uncertainty is suited to philosophy and religious domain. One important aspect of this classification is that, a given situation or phenomenon can contain several level of uncertainty at the same time. Therefore, this classification exists as a continuum or range of uncertainties rather than confined to a specific boundary. For further detail, please refer to Lo and Mueller's paper [155] which include two case study (one case study related to physics and the other to finance), where the taxonomy is being applied. Lo and Mueller's taxonomy is shown in Figure 3.12.

3.7.3.7 Conclusion to Classification of Uncertainty: Insights from other fields

These classification schemes offer useful categorization of uncertainty which enable researchers in each field to have a clear picture of the uncertainties to the scenario or phenomenon being encountered. Another useful application of classification is to enable the right treatment or solution to be used to reduce or tolerate uncertainty. On the other hand, there are issues that can be highlighted from the above discussion. Firstly, as mentioned at the beginning of Section 4.4, there are multitude of varying classification scheme being employed by researchers, either in the same or different fields of research. The drawback is the variation create confusion. From a positive point of view, different schemes might be agreeable since the differences are due to the fact that (1) there are different definition of uncertainty in the first place, and (2) different fields require different scheme due to different phenomenon being encountered. Secondly, different terms being used either in the classification or elements of the classes. For example, the term *dimension, form, view, class, environment, etc.* is being used to refer to classification. Subsequently, different terms such as *aleatory* vs *variability* or *endogenous* vs *external*, respectively carry the same meaning. Please refer to Table 3.6 for the summary of all the above classifications schemes.

For our purpose, these classification schemes offer a guideline to enable us to choose either to apply existing scheme (if suitable), make minor modification to suit our field of research (uncertainty in service provisioning), or create a new classification all together. The exercise for formulating the classification for uncertainty in service provisioning will be conducted in Section 4.4.

Table 3.6: Summary of Different Classification of Uncertainty

Term	**Classification**	**Description**	**Field**
generic	1. exogenous 2. endogenous	external uncertainty internal uncertainty	Product & System Design [146]
generic	1. external 2. internal	subject has no control subject has some control	Cognitive science & psychology (Howell & Burnett [144], Kahneman & Tversky [145])
category	1. epistemic 2. aleatory	gap in knowledge inherent uncertainty	Risk assessment[137], Reliability engineering [138, 139]
generic	1. known 2. unknown	variability in past cases unmeasurable differences in measurement	Engineering design process (Clarkson [141])
dimension	1. level 2. nature 3. location	range between determinism and indeterminacy epistemic & variability exists in the model complex	Decision support system (Walker et. al. [143])

Continued on next page . . .

Table 3.6 – continued from previous page

Term	**Classification**	**Description**	**Field**
type	1. conceptual model	qualitative description of the system with regards to occurring process.	Risk assessment (Zio & Apostolakis [140])
	2. mathematical model	the additional approximations & simplifications introduced to transform qualitative model into tractable mathematical expression	
	3. computer code model		
dimension	1. ambiguity	patient's self-evaluation of uncertainty in illness as vague or unclear.	Health care (illness), (Mishel [150, 151])
	2. complexity	different information that patients receive about treatment & care system	
	3. deficient information	inadequate information about patients' diagnosis	
	4. unpredictability	variability in patient's illness and outcomes	
dimension	1. complexity	due to multicausality, contingency, reciprocity, and unpredictability	Health care (Babrow et. al. [152])
	2. qualities of information	due to clarity, accuracy, completeness, volume, ambiguity, consistency, applicability, confidence in sources	
	3. probability	due to belief in specific or range of probabilities	

Continued on next page ...

Table 3.6 – continued from previous page

Term	**Classification**	**Description**	**Field**
	4. structure of information	due to order, and integration	
	5. lay epistemology	individual's belief about a phenomenon	
category	1. social integration	the reliability of social relationships related to disease's dynamics.	Health care (cancer related) (Kasper et. al. [153])
	2. diagnosis & prognosis	the current state of the disease and its future course	
	3. deciphering information	the interpretation of the behaviour of medical staff and other information receive by patients	
	4. mastering of requirements	the ability to cope with disease related life changes.	
	5. causal attribution	cognitive integration of being affected by a chronic disease	
	6. own preferred degree of involvement	the degree of active role (by patient) in patient-doctor relationship	
	7. physician's trustability	both professional and personal competencies of medical staff.	
	8. treatment	the efficacy of a treatment.	

Continued on next page ...

Table 3.6 – continued from previous page

Term	**Classification**	**Description**	**Field**
points of occurrence	1. context uncertainty	consists of endogenous & exogenous	Design process (Kreye et. al. [142])
	2. data uncertainty	consists of data incompleteness, inaccuracy, and variation	
	3. model uncertainty	consists of conceptual, mathematical, and computational	
	4. phenomenological uncertainty		
level	1. complete certainty	initial conditions are fixed & known, then all past & future states can be exactly determined.	Economics & finance (Lo & Mueller [155])
	2. risk without uncertainty	randomness is dictated by known probability distribution	
	3. fully reducible uncertainty	risk that has degree of uncertainty due to unknown probability	
	4. partially reducible uncertainty	a limit of uncertainty reducibility based on certain characteristics	
	5. irreducible uncertainty	total ignorance, cannot be solved by collecting more data or statistical inference	

3.7.4 Methods of Dealing with Uncertainty

There are several approaches in solving uncertainty based on the classification of uncertainty described above. Frequentist approach with traditional probability theory is used to analyse systems that are subject to aleatory uncertainty. Techniques such as Neyman-Pearson [156, 157] and Monte Carlo [158] are frequently used. On the other hand, epistemic uncertainty can be handled by several methods such as (1) possibility theory [159], (2) evidence theory [160], (3) Bayesian probability theory [161], (4) interval analysis [162], and so on. Bayesian methods are appropriate in situations where there are gaps in information (i.e. where there is epistemic uncertainty).

3.7.5 Relationship between Uncertainty, Risk, and Trust

There is a need to clarify the issue on the differences and relationship between uncertainty and risk since in common usage, both terms refer to a similar situation, in which some aspect of the future cannot be foreseen. In economics, the definitions of these two terms are different as established by Frank Knight in his book, *Risk, Uncertainty and Profit* in 1921 [130]. According to Knight, risk is present when future events with probability that is measurable whereby uncertainty is present when the likelihood of future events is indefinite or incalculable. Mathematically, the risk defined above can be expressed as follows:

$$\mathrm{R} = P(E) * C \tag{3.1}$$

whereby,
R = is the calculated risk
$P(E)$ = is the probability of the event occurs
C = is the impact associated with the event

Another way to look at the relationship between uncertainty and risk was presented by Doug Hubbard [163] as shown in Table 3.7.

From the point of view of service provisioning, we have already defined uncertainty as the gap in knowledge towards the service offering, and we can view risk similarly to Knight's view on risk, whereby risk is just a state of uncertainty where some possible

Table 3.7: Relationship Between Risk and Uncertainty

	Description	**Example**
Uncertainty	The lack of certainty, a state of having limited knowledge where it is impossible to exactly describe existing state or future outcome, more than one possible outcome.	If the outcome of SLA is not known (in advance), then you have state of uncertainty.
Measurement of Uncertainty	A set of possible states or outcomes where probabilities are assigned to each possible state or outcome - this also includes the application of a probability density function to continuous variables	You can associate the probability of the possible outcomes using some measures of forecasting or a calibrated probability assessment. Suppose that you quantify the uncertainty as a 90% of chance for the SLA to comply.
Risk	A state of uncertainty where some possible outcomes have an undesired effect or significant loss.	If the service required (with the attached SLA) important to complete a task, then there is risk since there is a 10% chance of SLA failure to comply and this would be undesirable.
Measurement of Risk	A set of measured uncertainties where some possible outcomes are losses, and the magnitudes of those losses - this also includes loss functions over continuous variables.	If there is an associated loss of $1000 in the event of SLA not complied, then you have quantified/measure the risk where there is a: 10% chance of losing $1000

outcomes have an undesired effect or significant loss. More importantly, is there a direct relationship between uncertainty and risk?

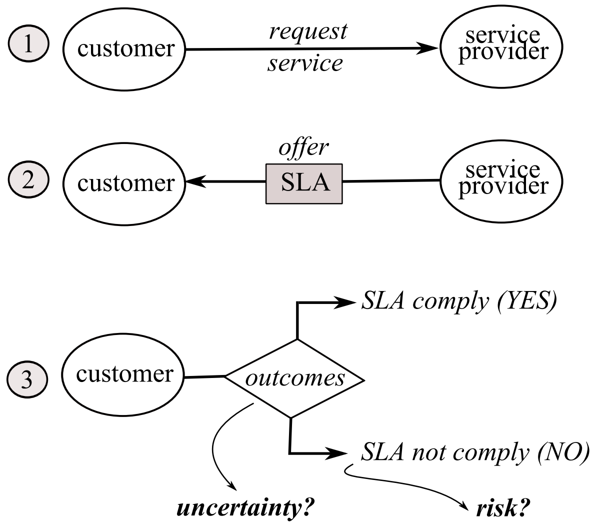


Figure 3.13: View on Risk and Uncertainty

Figure 3.13 illustrate the existence of uncertainty and risk within service provisioning offer. If the outcomes of the offer is not known in advance (which we view as uncertainty), and one of the outcomes has an undesired effect or loss (which we view as the risk), then it is possible to say that if we reduce uncertainty then we directly reduce risk.

How does trust comes into this picture in relation to uncertainty and risk? First, lets begin with a definition of trust. There are various definition in literature with regards to the definition of trust [43, 164, 165, 166], one of which we prefer is from Jøsang [167] which defines two type of trust: (1) **reliability trust**, and (2) **decision trust**. Reliability trust refers to the probability estimate of success of a transaction, whereby decision trust refers to the extent an entity disposition about entering into a transaction with another entity. We can use both definitions for our service provisioning offer scenario since the customer need to trust the service provider regarding the success of the transaction and also whether to accept the SLA offer from the service provider. If there is risk in the transaction, then we can say that risk affects the customer disposition or willingness to enter into a contract (through SLA).

Therefore, qualitatively, we deduce that the notion of trust is also linked with uncertainty and risk, whereby if uncertainty is reduced, then risk is also reduced thus increases trust. We will not pursue any quantitative relationship between uncertainty, risk, and trust since the objective of this discussion is to establish a qualitative relationship between these concepts.

3.7.5.1 Relationship between Trust and Risk

Trust is a relationship between two entities: (1) trustor (the trusting party), and (2) trustee (the trusted party). According to Jøsang et. al. [167], trust is the subjective probability of which the trustor expects that the trustee performs a given action on which its welfare depends. When the trustee performs as expected, then there is a positive outcome on the trustor, and vice versa. Therefore, the positive and negative outcomes corresponds to opportunity and risk. Thus, we can conclude that there is a relationship between trust and risk. It is often suggested that trust is inverse to risk, i.e. high trust means low risk. However, Kini and Choobineh [165] suggested that the relationship between trust and risk is not directly inverse proportional, and dependent on the stake of the outcome. Similarly, Solhaug et. al. [168] suggested that trust is generally neither proportional or inverse proportional to risk. Trust is inverse to the probability of risk and proportional to the value the trustee is willing to stake. His conclusion is that high trust only means low probability of a harmful incident.

The level of trust reflects the state of uncertainty about future behaviour of the trustee. According to Solhaug and Ketil [169], the uncertainty associated with trust can be both aleatory and epistemic. The aleatory uncertainty is associated with the inherent possibility of the trustee to be trustworthy and/or to deceive, while the epistemic uncertainty corresponds to the whether the trustor has access or means to get the evidence about the trustee.

3.8 Summary

This chapter provides the background for two important concepts which is services and uncertainty in general. Services are the foundation for service oriented computing paradigm and the nature of services (as compared to physical goods) presents an interesting challenge in term of quality guarantee. Furthermore, this chapter examines the issue of uncertainty from various other research areas in order to understand the diversity of this issue which leads to the current multitude of definition and classification approach. The background information in this chapter will be the basis for the discussion of the specific uncertainty issue in the area of service provisioning which is discussed in the next chapter.

Chapter 4

The Problem of Uncertainty and the Concept of Uncertainty Tolerance in Service Provisioning

4.1 Introduction

The issue of uncertainty in other research areas has been clearly defined and addressed in Chapter 3. Unfortunately, at present, there is no clear definition and classification of uncertainty from the context of service provisioning. The lack of clear definition and classification can cause misunderstanding among interested parties (researchers, system designer, system developer, etc.), misconception, and lack of focused effort on addressing the issue of uncertainty. In a way, to sum up the above problem, there is no existing abstraction on the issue of uncertainty within the context of service provisioning. Without this abstraction, it is difficult to derive a generic design for the uncertainty tolerance mechanism. This abstraction should contains the discussion and definition of various concepts that come into play when addressing the issue of uncertainty in service provisioning. Clarifying these concepts is a challenge, primarily due to the subjective nature of uncertainty.

This chapter presents the overall issue of uncertainty and introduces the overall concept of uncertainty tolerance in service provisioning with respect to service requester's perspective. There are three contributions in this chapter. Firstly, a classification scheme of uncertainty in service provisioning is presented. Secondly, a single view of uncertainty is presented based on the temporal classification scheme and the service requester's perspective. The third contribution of this chapter is the conceptual representation of uncertainty tolerance approach in service provisioning. To our knowledge, there is no existing work that conceptualizes the issue of uncertainty and uncertainty tolerance from the perspective of service provisioning. This conceptual model is useful to system designer and developer in taking account the issue of uncertainty and also potentially include aspect of uncertainty tolerance in the design. This chapter will also form the basis for the Uncertainty Tolerance Framework which will be presented in the next Chapter 6.

The remainder of this chapter is organized as follows: Section 4.4 presents the classification scheme of uncertainty in service provisioning based on several different dimensions. Section 4.5.2 presents a unique view of uncertainty in service provisioning based on the temporal classification scheme and also the service requester's perspective. The discussion in this section includes the justification for the view, justification for the chosen perspective, and the service requester's belief system. Section 4.6 introduces the concept of uncertainty tolerance which includes the discussion on the needs, degrees, and definition, plus a comparison with the concept of fault tolerance. Section 4.7 presents the generic concept of uncertainty tolerance, and finally Section 4.8 summarizes this chapter.

4.2 The Conceptual Model of A Service Provisioning System

This section presents a set of definitions that will be used throughout the discussion on uncertainty and uncertainty tolerance in this chapter and the remainder of the thesis. The scope of these definitions is limited to a generic service provisioning system. The aim is to provide a generic definitions that is able to cover the entire lifecycle of a service provisioning process and also applicable to different paradigm of service-oriented computing such as grid computing, cloud computing, and web services. Some of the definitions concerning services and uncertainty have been presented in general term in Chapter 3, but we will present those definitions in this section as part of the abstraction of the system model.

4.2.1 The Basic Concepts of A Service Provisioning System

A **service provisioning system** (hereafter is referred simply as **system**) is defined as a set of entities that interact and dependent on each other to achieve a specific goals, which are the provision and consumption of a service. A system is bounded by a **system boundary**, which provides the scope of the system. The system boundary can be in the form of (1) organizational boundary, whereby the system is limited within one or more organizations, (2) network boundary, such as internet (i.e. internetwork) or intranet, and (3) geographical boundary, limiting the system to certain country or locality (for example media streaming service such iPlayer[1] for BBC which is limited to UK users only). Other types of boundary can be defined base on the needs and goals of the system.

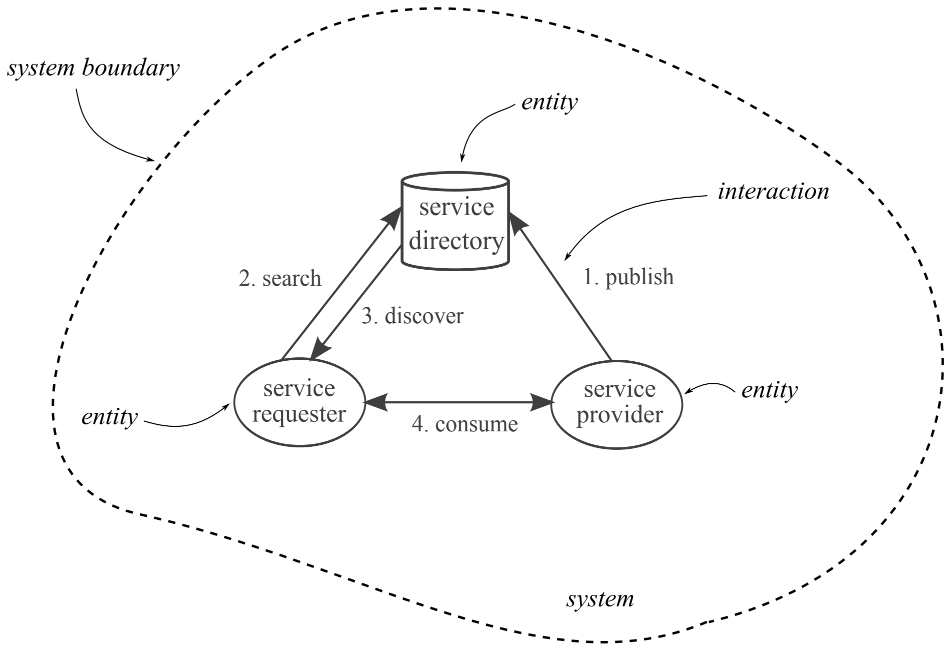


Figure 4.1: An Illustration of the System Model

An **entity** in a system can be a human or a software agent, and each entity has its own specific functions, and goals. In a generic service provisioning system, there are three entities as shown in Figure 4.1: (1) service requester, (2) service provider, and (3) service directory. The **function** of each entity in term what its intended to do is

[1]http://www.bbc.co.uk/iplayer/tv

determined by a specific functional specification. Each entity can have an interaction with one or more entities in a system. This **interaction** reflects an entity's action, whether to request or provide a certain task. Figure 4.1 illustrates the system, its boundary, the entities involved, and interactions between entities.

In the context of the system model, **service** is the central focus of the whole system. Each entity has a role or function related to service. A service provider functions to provision the service, the service requester functions as the consumer of the service, and the service directory functions as a metadata repository for service registration and discovery. As defined in Chapter 3, a **service**, from the context of service provisioning represents a relationship between at least one service provider and one service requester to achieve a specific business goal.

An extended model of the above system could include an additional entity, known as a *service broker* or *service mediator*. The function of the service broker is to negotiate on behalf of the service requester, when there are more than one service providers providing the same type of service. Other functions of service broker in the context of service provisioning are as advertiser (for the service providers), auctioneer, translator, and negotiator, as discussed in [170]. In theory, from the context of this research, it is possible to apply the proposed uncertainty tolerance mechanism (in Chapter 6) to cater for the situation whereby there are more than one service providers and mediated by a service broker. As an example, the service broker can use the uncertainty tolerance mechanism to select a service provider (from a set of service providers) on behalf of the service requester. The extended model of the system is shown in Figure 4.2. Due to limitation in term of scope and time, we propose that the implementation of proposed uncertainty mechanism for the extended system model is done as future work, following the completion of this research.

4.2.2 The Lifecycle of A System

A system consists of a lifecycle with distinct phases. A **phase** is defined as a distinct period or stage in the lifecycle. Each phase within the lifecycle contains activities. An **activity** in a phase is defined as a specific action taken by one or more entities to accomplish a specific task. To simplify the lifecycle and to make it as generic as possible to cover the different paradigms of service-oriented computing, only important

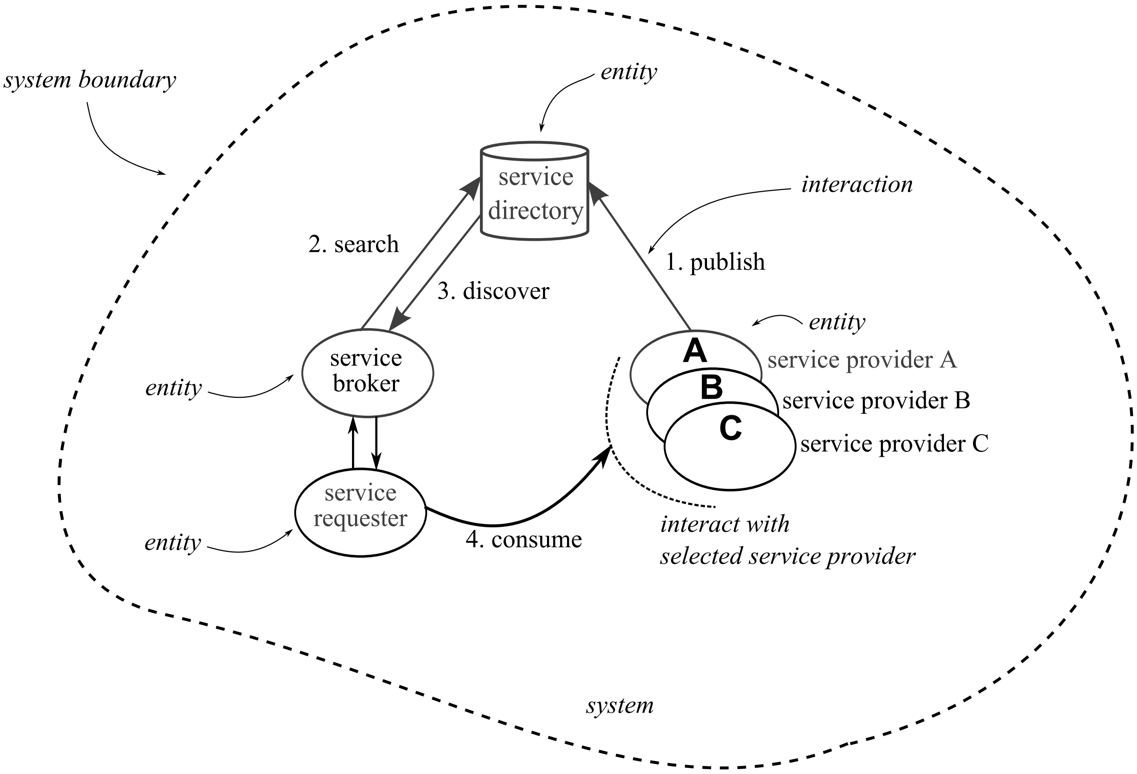


Figure 4.2: An Illustration of the Extended System Model

activities are discussed in this lifecycle. The lifecycle of a service provisioning system consists of three phases (as shown in Figure 4.3): (1) pre-sp[1], (2) intra-sp, (3) post-sp.

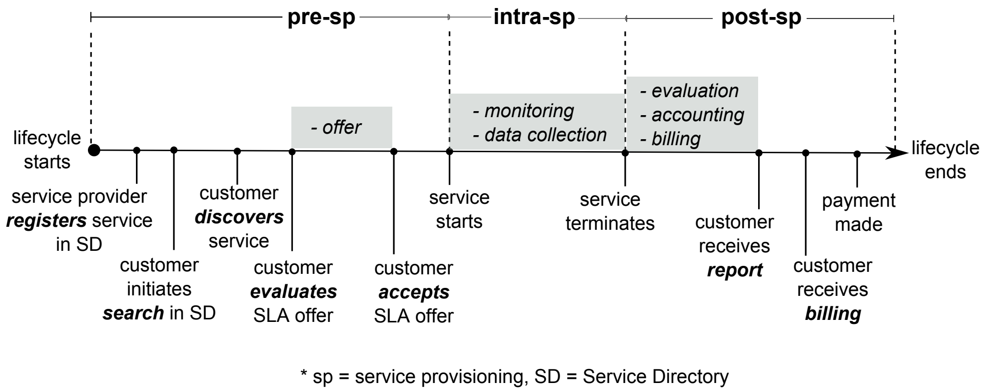


Figure 4.3: An Illustration of a System's Lifecycle

pre-sp: The pre-sp phase takes place prior to the actual service provisioning pro-

[1]service provisioning

cess. In this phase, there are several distinct activities between the entities. The first activity is the **register** activity which involves two entities, a service provider and a service directory. A service is first registered in a service directory by a service provider. The registration activity enables the information about the service (metadata) to be stored in the service directory, for the purpose of service discovery by service requesters. The quality of the service is guaranteed by the service provider through a Service Level Agreement (SLA), which is part of the metadata registered by the service provider. A customer (we use the term customer interchangeably with service requester) initiates a **search** activity in the service directory to find a suitable service to achieve a specific business goal. When a suitable service is found (by the customer), this activity is termed as **discover**. At this point, the customer has to undertake an important activity which is to evaluate the SLA **offer** *linked* to the service. In real world scenario, a potential implementation on linking the SLA and service description is through UDDI (Universal Description Discovery and Integration) standard (as discussed in Chapter 3 , Section 3.3.4.1). For example, Rajendran and Balasubramanie [171] proposed the usage of ***tmodel*** in UDDI resgitry for a service to provide the SLA information or to point to the actual SLA document at the service provider location. Therefore, it is possible to link between service discovery with relevant SLA for that particular service using *tmodel* approach.

We use the term **offer** to indicates that the service provider is actually extending an invitation to the service requester to accept the service guarantee. The evaluation process of the offer by the customer is subjective in nature since it depends on many factors such as customer's previous experience, customer's knowledge (about the service, and service provider), etc. The customer then makes a decision to accept or reject the **offer**. The pre-sp phase ends when the service consumption starts.

intra-sp: The intra-sp phase refers to the period to when the service provisioning process starts and ends. The main activity that interest us in this phase is the monitoring activity. **Monitoring** refers to the act of observing the specified metrics over a period of time. It involves the service provider to collect specific data at certain interval or during a specific process.

post-sp: The post-sp phase refers to the period after the service has been terminated or completed. There are two main activities of interest: (1) **Evaluation**, and (2) **Accounting**. Evaluation refers to the process of analysing the previously collected data while accounting refers to the process of generating an invoice by looking at the resource consumption and calculate how much is the associated cost.

4.2.3 Service and Service Quality

The quality of service (QoS) plays an important role in a service provisioning system. From a business perspective, a service bounded by a QoS provides a measure of guarantee and a sense of confidence for a service requester. Additionally, QoS can be a criteria to choose service offers from a set of a service providers that offer similar service.

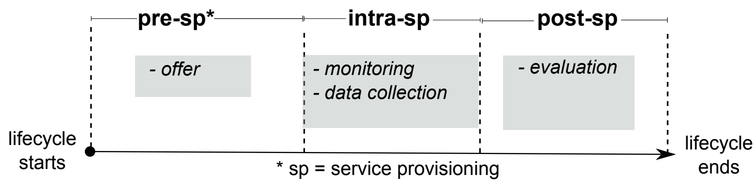


Figure 4.4: An Illustration of a System's Service Quality Activities

Based on Figure 4.4, there are four main activities within a service provisioning lifecycle related to the quality of service: (1) SLA offer, (2) monitoring of service, (3) data collection (related to service metric(s), and (4) evaluation of service. We can see that (1) is in the pre-sp phase, (2) and (3) are in the intra-sp phase, and (4) in the post-sp phase.

4.3 The Threat to Service Provisioning: Uncertainty

From the perspective of a service requester and service provider, a service needs to fulfil its objective and comply with the agreed service guarantee (SLA). Uncertainty throughout a service provisioning lifecyle can affect both the service requester and service provider. This section will provide a discussion on the

4.3.1 Definition of Uncertainty

To the best of our knowledge, there is no existing definition of uncertainty within the context of service provisioning. We have compiled a list of definition of uncertainty from different fields in Chapter 3 before we come up with our own definition of uncertainty from service provisioning point of view. The definition of uncertainty from the context of service provisioning is as follows:

Definition 5 *(Uncertainty): The gap in knowledge or lack of information in the service provisioning which affect customer's degree of belief and cause difficulty in decision making (from the customer point of view).*

The gap in knowledge can be caused by i) absence or lack of data, ii) unknowns about the source of data, and iii) inherent uncertainty (as in physics and statistics).

4.4 Classification of Uncertainty in Service Provisioning

To our knowledge, there has been no attempt to provide a classification scheme for the problem of uncertainty within the context of uncertainty tolerance. From the context of our work (uncertainty tolerance), a classification scheme is defined as a system of organizing or arranging different uncertainties within a service provisioning system into groups or types, based on one or more criteria. The lack of consistent classification scheme can cause two problems: (1) since the type of uncertainty can be linked to different treatment or solution, the lack of proper classification scheme can lead to unsuitable treatment (less efficient or accurate), (2) lack of consistent classification can lead to poor understanding of the underlying problem. Previous classification schemes for other research fields, which have been discussed in Chapter 3 will serve as a guideline for our classification scheme.

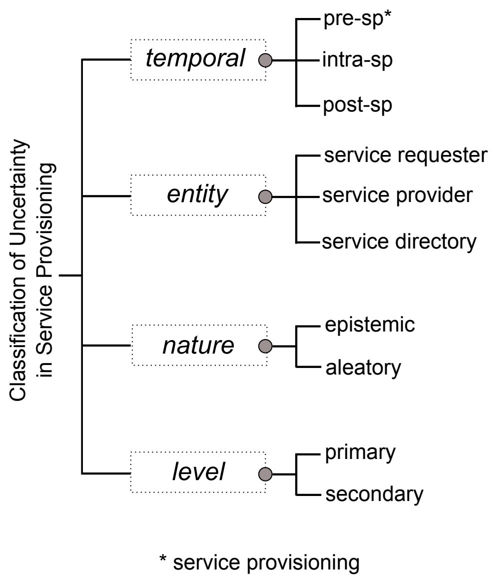


Figure 4.5: Classification of uncertainty in service provisioning

The generic architecture of service provisioning provides information on the ***entities*** involved, and the relationship between them. We can also view entities from spatial aspect referring to the *location* of the uncertainty. Furthermore, the generic lifecycle provides information on the ***temporal*** aspect of the uncertainty (in term of phases within the lifecycle) and processes or actions that occur during the lifecycle. Additionally, two other classification dimensions are considered, the ***nature*** and the ***level*** of the sources. The ***nature*** dimension consists of empirical and aleatory categories used by various engineering fields classification and also in Walker et. al. classification will be adopted as well. The ***level*** category consists of primary and secondary level of uncertainty. To conclude, for uncertainty in service provisioning, there are four main categories (entity, temporal, nature, and level), each with further sub-classes (as shown in Figure 4.5). Each of these classifications category and sub-classes will be discussed in detail in the following sections.

4.4.1 Temporal Classification of Uncertainty

It is possible to classify uncertainty from different temporal view within the lifecycle of service provisioning. Having these different views enable us to understand how and which aspect of service provisioning is affected. In this classification scheme, the service provisioning process is divided based on temporal constraint. There are three distinct temporal-phase within a single service provisioning transaction (as illustrated in Figure 4.6): (1) pre-sp[1], (2) intra-sp, and (3) post-sp. This classification scheme will be the basis for presenting uncertainty in service provisioning.

4.4.2 Entity Classification of Uncertainty

The temporal-phase view above does not specifically include the entities involve in a service transaction and the relationship between the entities. Knowing the entities involve and the relationship between them can be useful, for example, if there is uncertainty in a particular activity (within the service provisioning), it is possible to identify the entity that should resolve the problem. In a generic and simplified example of a web-based service provisioning, we assume that there are three main entities involved in a service transaction as shown in the following figure.

[1]sp: service provisioning

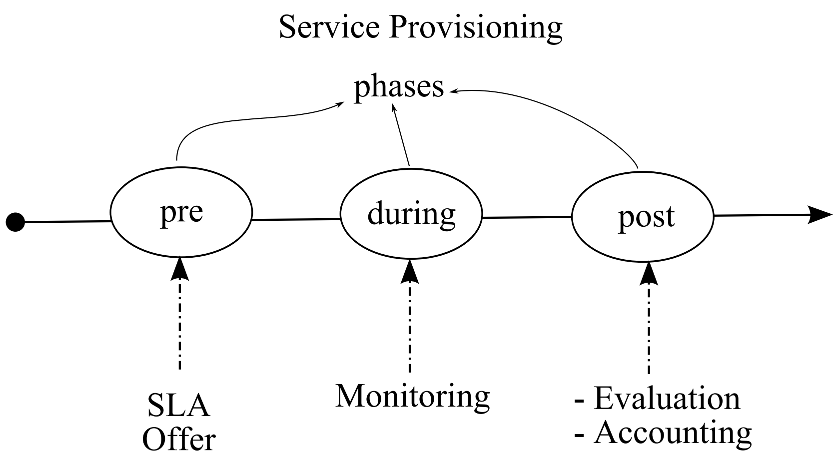


Figure 4.6: Phase View of Uncertainty

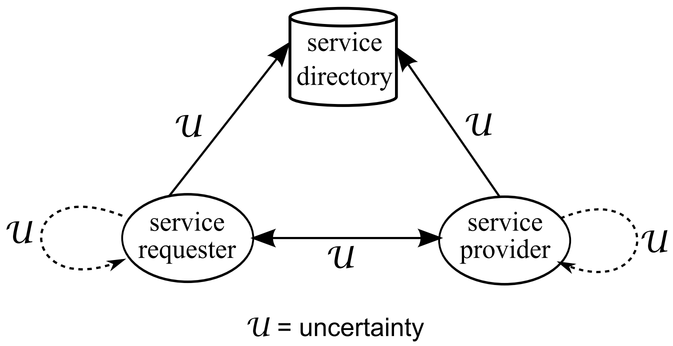


Figure 4.7: Entity View of Uncertainty

The three entities are (1) service requester, (2) service provider, and (3) service directory. In each of these entities, uncertainty can arises as a matter of subjective belief towards another entity or to itself. For example, a service provider might have uncertainty towards its capability to deliver or fulfil the quality promise within the SLA and a service requester can have uncertainty towards the service provider capability. As for the service directory, if it also functions as a trusted third party to monitor and collect data, both the service requester and service provider might have uncertainty in term of the trustworthiness of the third party.

4.4.3 Nature Classification of Uncertainty

Similar to the classification scheme used in engineering related fields (Section 3.7.3.1), uncertainty in service provisioning can also be classified based on the ***nature*** of the uncertainty, either epistemic or aleatory. To recap, epistemic uncertainty refers to the gap in knowledge or lack of information of a certain situation, and aleatory uncertainty is due to the inherent variability in a system or process. As an example, uncertainty in customers's degree of belief is considered epistemic since the uncertainty is caused by lack of information about the service offer probability to be complied to. On the other hand, uncertainty in measurement of the metrics (such as latency, uptime, etc.) is aleatory in nature since there is randomness or inherent variability due to physical limitation (law of physics) of the system and accuracy of measuring device.

4.4.4 Level Classification of Uncertainty

The ***level*** classification of uncertainty in service provisioning refers to the degree of relationship between the uncertainty and the *system* itself, whereby the term *system* refers to the service provisioning environment. This *level* classification contains two type: (1) primary uncertainty: uncertainty which is directly originated or located within the system, and (2) secondary uncertainty: uncertainty which is not directly related or located within the system. In a way, this classification types are similar to *exogenous* (external) and *endogenous* (internal) classification scheme discussed in Section 3.7.3.2. As an example, the service requester's degree of belief towards an SLA offer from the service provider is considered primary uncertainty since it is directly originated within the system while trustworthiness of evidence used to update service requester's initial belief is considered secondary uncertainty since it is not directly related to the system (uncertainty of the evidence).

4.4.5 Multidimensional Classification of Uncertainty

Although we have defined four separate types of uncertainty in service provisioning, in reality (with respect to service provisioning), uncertainty can be classified into more than one type. This is similar to the approach suggested by Walker et. al. [143] (Section 3.7.3.3) which states that uncertainty is a three dimensional concept. For example, service requester's degree of belief towards an SLA offer can be considered *epistemic*, *primary*, *service requester entity*, and within the *pre-sp* phase.

The illustration for multidimensional classification for SLA offer is shown in Figure 4.8. This hybrid classification scheme provides a much more wider scope of classification to interested parties such as researchers, system designer, and system developer.

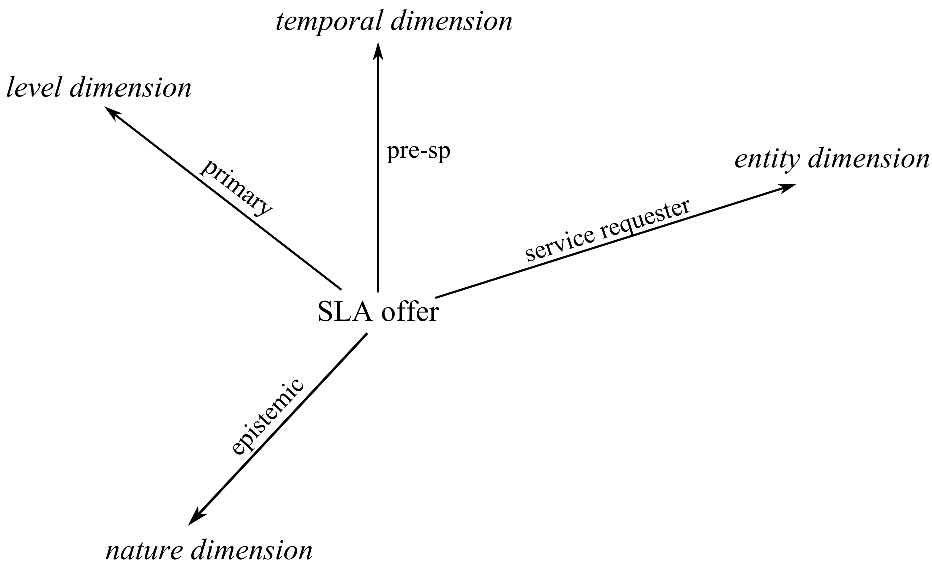


Figure 4.8: An Illustration of Hybrid Classification for SLA Offer

A (non-exhaustive) list of uncertainty in service provisioning and its related hybrid classification is provided in Table 4.1.

Table 4.1: List of Uncertainty in Service Provisioning and its Hybrid Classification

		Classification				
Uncertainty	**Description**	**temporal**	**entity**	**nature**	**level**	**Action**
service requester's belief towards SLA offer from service provider	uncertainty due to the fact that the state of the event (SLA compliance) in the future cannot be determined in advanced	pre-sp	service requester	epistemic	primary	need to be tolerated
data collection and monitoring for metrics	(1) limits in the accuracy and precision of the tools (apparatus and/or software)	intra-sp	service provider	aleatory	primary	can be ignored
	(2) service requester's belief towards the trustworthiness of the monitoring and data collection process.	intra-sp	service requester	epistemic	primary	need to be tolerated
validation and accounting	service requester's belief towards the trustworthiness of the validation and accounting process.	post-sp	service requester	epistemic	primary	need to be tolerated
evidence	uncertainty in the evidence used to update the initial service's belief	pre-sp	service requester	epistemic	secondary	assumed trusted

4.5 Using Temporal View to Describe Uncertainty in Service Provisioning

The classification scheme provided in the previous section is useful in justifying the different type of uncertainties that can exist in a service provisioning system. The challenge is, depending on the situation and the view taken by a person trying to describe the uncertainty from the context of service provisioning system, any approach to represent uncertainty using the classification schemes described in previous section can be used. Hence, it is possible to have different types of view depending on the person who is describing it. Unfortunately, this could potentially leads to confusion among interested parties.

Therefore, this section aims to provide one possible approach to represent the uncertainty in the context of service provisioning from temporal classification scheme. Furthermore, the proposed representation will enable us to develop a suitable uncertainty tolerance mechanism in line with the representation.

4.5.1 Justification

The choice of using temporal classification scheme as the basis for representing uncertainty in service provisioning context is based on the following justifications:

- **completeness**: using temporal view and inclusion of the complete lifecycle of a service provisioning process enable the complete representation of uncertainty. Other classification schemes such as the *nature* based or *entity* based does not provide a complete coverage of the uncertainties.

- **process based**: the temporal view of uncertainties consider processes within the lifecycle of a service provisioning as the basis to justify the point of origin for the uncertainty issue. Using process as the point of origin is a logical step since it provides context to the uncertainty issue being discussed.

- **logical flow**: the temporal view using the lifecycle provides a logical flow from a starting point to an end point. As such, this approach is easier to explain and understood by interested parties.

4.5.2 Uncertainty in Service Provisioning: A Temporal View

The basis for temporal view approach of presenting uncertainty in service provisioning is by using the lifecycle of a service provisioning process. This lifecycle should be as generic as possible to cover different paradigm of service-oriented computing such as web services, utility computing, grid computing, and cloud computing. As such, the generic lifecycle does not fully covers all possible processes within a service lifecycle. For example, negotiation, which refers to the process by which the parties involved (in this case the service requester and provider) come to a mutual agreement [172, 173], is not considered in this generic lifecycle. A generic lifecycle with associated processes is shown in Figure 4.9.

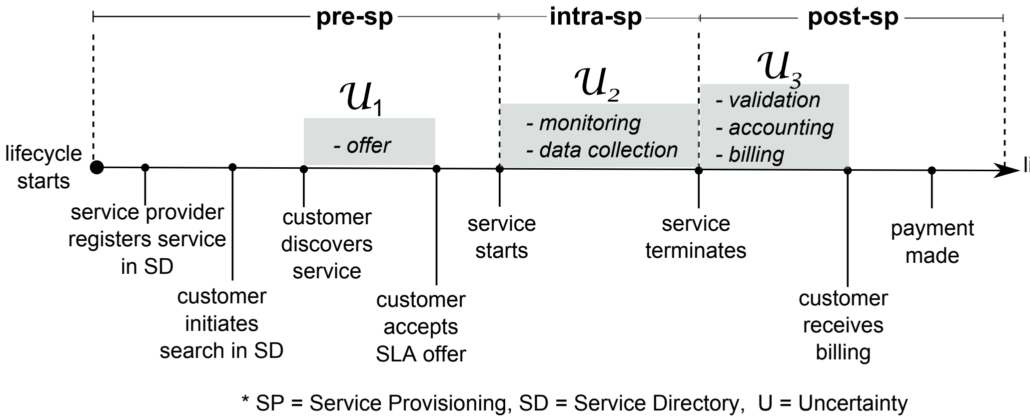


Figure 4.9: Temporal View of Uncertainty

pre-sp: The pre-sp phase takes place prior to the actual service provisioning process. In this phase, there are several distinct activities based on the service provisioning lifecycle and possible associated uncertainty. One activity of interest to us within this phase is the SLA offer from the service provider to the customer (assuming the service provider has registered the service with associated SLA and the customer has conducted the search and discovery of required service). We define this scenario as the subjective uncertainty from the viewpoint of the customer to the probability of the SLA being complied to. The cause of the this uncertainty can due to the lack of information or gap in knowledge from the customer point of view about mainly the service provider, possibly in term of its past performance record, and other factors. Thus, this scenario can be clearly placed under the *epistemic uncertainty* classification.

intra-sp: The intra-sp phase refers to the period to when the service provisioning process starts and ends. The main activity that interest us in this phase is the monitoring activity. *Monitoring* refers to the act of monitoring the specified metrics within the SLA to check with compliance and violation. It involves the service provider to collect specific data at certain interval. We view the monitoring and data collection as another sources of uncertainty. Furthermore, it is possible to classify two type uncertainty for the same source, i.e. from the monitoring and data collection activities. The first source of uncertainty is related to the limit of accuracy and precision of the equipment or apparatus being used to measure and collect the data. This source of uncertainty can be classified under the aleatory group. Often, the uncertainty of a measurement is found by repeating the measurement sufficient number of times to get an estimate of the standard deviation of the values. One possible measure that can be taken to tolerate this type of uncertainty is to have a better accuracy and precision of the measuring instruments. The second source of uncertainty is the subjective belief of the customer towards the trustworthiness of the service provider or the party that conduct the monitoring and data collection activity. Furthermore, this state of belief (or disbelief) is caused by the lack of information or gap in knowledge towards the party that conduct the monitoring and data collection activity. This particular source of uncertainty is of interest to us for this research and due to the nature of insufficient information of the uncertainty, we can classify it under the epistemic uncertainty group.

post-sp: The post-sp phase refers to the period after the service has been terminated or completed. There are two main activities of interest: (1) **Evaluation**, and (2) **Accounting**. Evaluation refers to the process of analysing the previously collected data while accounting refers to the process of generating an invoice by looking at the resource consumption and calculate how much is the associated cost. The nature of uncertainty of both processes is *epistemic* since we can view this similarly to previous phases whereby the customer has subjective belief towards the trustworthiness of the service provider to use the correct accounting model and also proper evaluation criteria and method. Additionally, there is also an issue of *trust uncertainty* whereby the customer assumes that the service provider always report the truth about the outcome of the validation [174].

Based on the above discussion, the uncertainty in service provisioning can be clearly explained using temporal approach through the lifecycle of a service provisioning process. The list of processes in the lifecycle is not exhaustive to cover all possible processes as discussed in Chapter 3, since the objective of this section is to provide a starting point or groundwork for describing uncertainty in a generic service provisioning environment. Furthermore, we only consider uncertainties which are critical to the service provisioning process and can cause negative consequence to the service requester.

4.5.3 Who's Perspective?

In order to provide a focused description of the uncertainty, the temporal view of uncertainty in service provisioning discussed in the previous section is based on the service requester's (i.e. customer) perspective, as shown in Figure 4.10. This distinction is needed since there are different types of entity in a service provisioning environment as discussed in Section 4.4.2, and the uncertainty can affect each entity in a different way. Furthermore, the perspective of the uncertainty will be an important factor in defining the concept of the uncertainty tolerance which will be discussed in coming section.

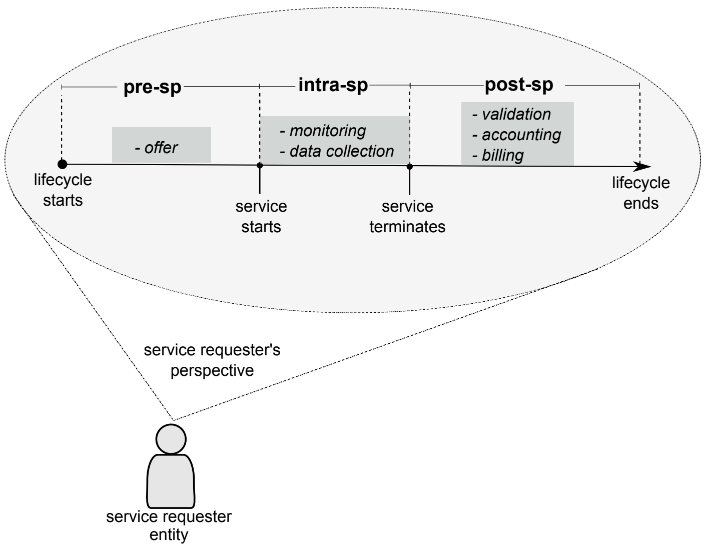


Figure 4.10: Service Requester's Perspective of the Uncertainty

Based on the temporal view approach, combined with the service requester's perspective, the uncertainty in service provisioning can be linked to the belief system of the service requester. For example, when presented with a service offer from the service provider, the initial degree of belief of the service requester towards the probability of compliance of the offer is viewed as uncertainty. Essentially, this belief is the underlying foundation of the uncertainty issue in the service provisioning system that this thesis is addressing. Table 4.2 summarizes the uncertainty in the service provisioning lifecycle based on the temporal view and service requester perspective plus the belief statement that constitute the uncertainty.

Table 4.2: Summary of Critical Uncertainty in Service Provisioning with Belief Statement

Phase	**Process**	**Belief**
Pre-sp	offer	*belief* towards the compliance of the offer.
Intra-sp	monitor	*belief* that the monitoring is correct and accurate.
Post-sp	report	*belief* that the report is truthful.

The following section provides justification on how a service requester can arrive at an initial degree of belief.

4.5.4 Justification for Service Requester's Initial Degree of Belief

In the scope of this thesis, we are not investigating or including the steps or concrete process of how a service requester arrives at their individual's degree of belief towards an SLA offer. Instead, we will provide justification on possible ways how service requesters' belief is formed. The justification of how a service requester (i.e. customer) arrives to an initial degree of belief should be applicable to different processes and belief statement presented in Table 4.2. The following discussion however uses the *"offer"* uncertainty as a case study for the justification of the initial degree of belief.

Customers' perception towards service quality, as discussed in Chapter 3, has been a long debated issues in economics. To recap, the challenge in building the perception of quality for consumption of service as compared to the consumption of goods lies in the fact that there is lack of tangible physical cues with regards to a service. In relation to our work in uncertainty tolerance, the same challenge applies when a customer is presented with a service quality proposition (in the form of SLA), how does the customer's perceive the compliance of the service? Again, it is important to restate that this perception is absolutely subjective and differs between two or more customers. However, there must be a valid justification for the formation of individual belief.

Lets define the process of forming a customer's belief as a ***belief bootstrapping*** process. The term ***bootstrap*** has been chosen to represent the cognitive process of a service requester that receives input, processes the input, and produces the initial belief. The bootstrapping process involves two streams, (1) input stream, and (2) output stream, and a bootstrapping process node in the middle as shown in Figure 4.11. The *input* stream represents a channel or pathway of which various information from potentially various sources regarding the subject in question being streamed into the *bootstrap processing* component. The *initial belief* is the outcome of the *bootstrap processing* module. The bootstrapping processing module receives the input and go through certain procedure to process the input and produce the output. We assume that the actual mechanism of bootstrapping process is based on human cognitive mental process. The specific of such process is not of interest to this thesis. More information about related work on bootstrapping is available from [175].

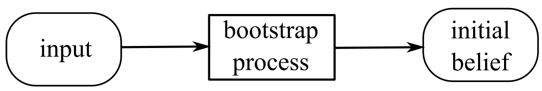


Figure 4.11: Initial Belief (prior) Bootstrapping Process

There are several possible sources of input, and can be categorized as internal and external inputs. Internal input is a type of input whereby the source is from within the customer itself, for example, customer's own experience of using the same service in the past. External input is the opposite of the internal input, whereby the source is external to the customer itself. As for external input, one such example is information pertaining to the service or service provider itself which might be included in the SLA or part of customer knowledge. For example, one such information is probably the name or *branding* of the service provider. Given two service provider, one is from a well known company such as Amazon and another is *"unknown"* company XYZ, a customer might perceive a service offer by Amazon have a higher chance to comply as compared to XYZ. Again, this belief is individualistic and subjective (consistent with our earlier discussion on subjectivity of the prior) in nature since two different customers might have different prior knowledge about these companies in the first place. Another useful information is the cost of the service. Although price usually indicates cost, it can also infers the level of quality [176, 177, 178]. For example, if a service has a very low price, which is not proportionate to the cost of the service, this might indicate a low probability of compliance for the SLA. The belief bootstrap process with different type of inputs is illustrated in Figure 4.12.

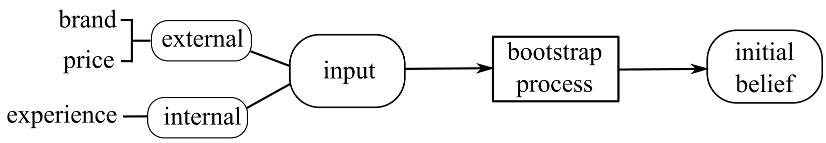


Figure 4.12: Initial Belief (prior) Bootstrapping Process with Different Type of Inputs

To summarize, the objective of the above discussion is to justify how a service requester can form an initial belief towards service provisioning activities which are linked to uncertainty such as an SLA offer from a service provider. Although under the subjective probability theory, an individual can assign any arbitrary value for the degree of belief (the prior), we have provided a justification on how this belief is formed based on a systematic process.

4.6 The Concept of Uncertainty Tolerance

This section introduces the concept of uncertainty tolerance with respect to the uncertainty issue in service provisioning environment.

4.6.1 The Needs for Uncertainty Tolerance

Previously, in Section 4.5.2, we have clearly identified the various types of uncertainty that can occur within the lifecycle of a service provisioning. We have identified that the uncertainty faced by a customer when presented with an contractual offer (in the form of an SLA) from a service provider is the customer's subjective belief of the uncertain state of the outcome of the offer. This is due to the fact that the customer does not have sufficient information or any information to confirm its own believe.

In order to justify the needs for uncertainty tolerance, we will present two possible scenarios in relation with service provisioning environment.

4.6.1.1 1st Scenario: Uncertain Customer's Initial Degree of Belief

Even though an individual is allowed to assign an arbitrary value for the initial belief, that value does not necessarily reflects the actual probability of the event occurring. Although this is understandable (the inaccuracy is simply due to the fact that the individual does not have enough information to make accurate judgement), such uncertainty can cause negative consequence if the decision making process includes the individual's initial subjective probability (i.e. the prior). Therefore, if a customer assign a belief to an offer, and if the belief deviates from the actual value, then the action of the customer (whether accepting or rejecting the offer) can cause negative consequence. Negative consequences can include loss of utility and loss of potential financial gain if the service succeeds (when the offer is rejected based on initial belief).

4.6.1.2 2nd Scenario: Uncertain Customer's Initial Degree of Belief with Multiple Offers

The second scenario involve a single customer receiving similar offer for a similar service, but from different service provider. Assuming that the customer does not have any additional information whatsoever regarding both service providers, it is likely that the customer's initial degree of belief (the uncertainty, i.e. the prior) would be the same for both offers. If the actual probability of success of the offer is the same, then there is no adverse affect if the customer accepts any of the offer. On the other hand, if the actual probability of success of one of the offer is greater than the other, then selecting the wrong one would bring adverse consequence to the customer (for example loss of utility).

The above two scenarios provide sufficient motivation to provide some form of tolerance towards uncertainty in service provisioning offer. Furthermore, apart from the above negative consequences, the existence of a mechanism to tolerate uncertainty would certainly affect customer's confidence, i.e. increasing the customer's confidence.

4.6.2 Degree of Uncertainty Tolerance

The degree of (service provisioning) uncertainty is not absolute. As such, this indicates that the degree of uncertainty tolerance is also not absolute. First of all, there are two main reasons why the degree of uncertainty is not absolute:

1. **subjectivity**: as stated before, the nature of the uncertainty is subjective, two different customers can have totally different belief towards the same service offer. As such, the deviation of their belief and the actual truth can be considered as different degree of uncertainty. Since the uncertainty tolerance mechanism includes the subjective initial belief in the uncertainty tolerance process, the degree of uncertainty tolerance is also dependent on the *subjectivity* of the initial belief.

2. **context**: contextual reason can be divided into two different type. First, a situation where two different customers using the same service, from the same service provider, but try to achieve a different goal. For example, for a data storage service, customer A might be using the service to store information which is not critical to the customer, and for customer B, it is the opposite situation (data is important). In this case, the degree of uncertainty is affected by the context of the goal or the objective of the customers. Secondly, the context of the importance of the service itself. For example, services, intended for financial sectors has higher stake as compared to services for leisure sectors. Therefore, the degree of uncertainty can also be different, taking into consideration the importance of the targeted sectors.

Therefore, since the degree of uncertainty is not absolute, then logically the degree of uncertainty tolerance cannot be absolute. Based on the above two reasons, we can conclude, for different customers, the degree of uncertainty tolerance is affected by individual subjectivity, and the context of the service.

4.6.3 The Definition of Uncertainty Tolerance

Our approach to tolerate uncertainty in service provisioning lifecycle is directly related to the underlying fact that the uncertainty is caused by the lack of information or gap in knowledge.

As discussed in Chapter 3, uncertainty is a complex subjective term. Our work on uncertainty focuses on the *subjective uncertainty* faced by a customer in the pre-sp phase, in particular when the customer is presented with service offer. Thus, our definition of uncertainty tolerance reflects the scope of uncertainty that we address. We choose the term *"tolerance"* since the term reflect the nature of uncertainty which is usually not possible to be removed entirely. Therefore, with respect to the customer's subjective uncertainty, the definition of uncertainty tolerance is as follows:

Definition 6 *Uncertainty Tolerance refers to the process that is able to reduce or minimize the gap between initial belief and the actual belief of an individual service requester, so that the effect of negative consequence of uncertainty can be minimized.*

The above definition is written to be as generic as possible in order to be applicable to the uncertainties that exist across a service provisioning lifecycle. Where necessary, a much more detailed definition can be defined for specific uncertainty within the service provisioning lifecycle.

4.6.4 Uncertainty Tolerance vs Fault Tolerance

The usage of the term *"tolerance"* in *"uncertainty tolerance"* inadvertently bring comparison to the area of *fault tolerance*. Is there any similarity between *uncertainty tolerance* and *fault tolerance*? To answer that question, first we will briefly discuss fault tolerance in computing system. In general, fault-tolerant system is defined as the ability of the system to function normally under the presence of errors or faults, with the objective to achieve dependable computing [179]. General approaches in achieving a fault-tolerant system is by employing redundancy. In one of his paper related to fault-tolerant computing, Avizienis [180] defined two types of redundancy: (1) hardware redundancy (consists of components introduced to provide fault tolerance), and (2) software redundancy (all additional programs/instructions which would not be needed in a fault-free computer). In hardware redundancy, we can differentiate between *redundancy* and *replication*. Using replication, several identical instances or components operate concurrently and a voting (quorum) system to select the correct outcome. On the other hand, using redundancy, there are several identical units or components but only one is operating at any one time, while the other unit(s) are standing by to take over in the event of failure.

Based on the above information about fault tolerance, and the definition of uncertainty tolerance in previous section, the term *tolerance* in both area loosely refer to the same objective, i.e. the system can function normally under the presence of unwanted factors (fault or uncertainty). On the other hand, the approach taken in uncertainty tolerance (which will be discussed in details in following sections) differs from the approach taken in fault tolerance. In essence, no notion of redundancy or replication is being used in uncertainty tolerance, rather the approach is to reduce the gap in knowledge through evidence gathering.

4.7 A Generic Concept of Uncertainty Tolerance in Service Provisioning

4.7.1 Requirements

- **belief system**: the concept must acknowledges that the service requester has a belief system that leads to the initial degree of belief towards the service provisioning offer. The belief of a service requester (i.e. a customer) can be a strong factor that affect the choice of action whether to accept or reject the offer from a service provider. Allowing a service requester to be able to assert its belief in the process is a powerful indicator that the service requester is involved in the process not just a bystander and agrees on what is offered by the service provider.

- **increased confidence**: although confidence is not measured quantitatively, the outcome of the uncertainty tolerance should increases the confidence of the service requester towards the service offer from the service provider. The term *confidence* can be interpreted as the the state of feeling about the truth of something. In the context of this thesis, confidence refers to the state of feeling of the service requester about the compliance of the service offer from the service provider. In a sense, it is the opposite of the term *uncertainty*. Figure 4.13 illustrates the relationship between uncertainty and confidence. Qualitatively, the relationship between uncertainty and confidence can be stated as follows: reducing uncertainty will increase the confidence of the entity in concern which is the service requester towards the compliance of the service offer, and vice versa. Wesson [181] and Peterson et. al. [182] have shown that there is a correlation between uncertainty and confidence level.

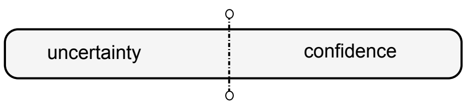


Figure 4.13: An Illustration of The Relationship Between Uncertainty and Confidence

- **positive outcome**: the uncertainty tolerance concept should produces a positive outcome for the service requester in term of the updated belief and other quantitative measure such as utility value. For example, the updated belief (i.e. tolerated uncertainty) should be closer to the actual truth if the original belief value deviates from the actual value. As for the utility value, a positive outcome should be a positive increment of cumulative utility value to the service requester. As for uncertainty in service provisioning offer, the positive outcome can also be in the form of the the service requester ability to make decision whether to accept or reject the offer, based upon the combination of the tolerated uncertainty and utility theory. Without the uncertainty tolerance mechanism, service requester might have to relies on gut instinct, incomplete information, or simply randomly accept the offer.

4.7.2 The Means to Attain Uncertainty Tolerance

This section will discuss in general the means to attain uncertainty tolerance in a generic service provisioning process from the perspective of a service requester.

uncertainty detection: in the context of the uncertainty in service provisioning from the perspective of a service requester, there is no need for a detection mechanism. This due to the fact that, uncertainty, in the form of service requester's belief is always present. For example, given a service offer from a service provider, a service requester, through the bootstrapping process discussed in Section 4.5.4, can assign a subjective initial degree of belief. Since this belief, in principle, cannot accurately predict the actual state of the future, therefore uncertainty will always be present in such situation.

uncertainty tolerance technique: Essentially, the means to attain uncertainty tolerance in service provisioning is linked to the characteristics of the uncertainty being discussed in previous section, which are: (1) based on service requester's perspective, (2) linked to service requester's belief system, and (3) epistemic (i.e. gap in knowledge). Therefore, any mechanism or approach that can fulfil the above characteristics can be considered as a potential solution.

Figure 4.14 illustrates the overall concept of uncertainty tolerance. The uncertainty tolerance concept consist of several components as shown in the figure. The ***initial***

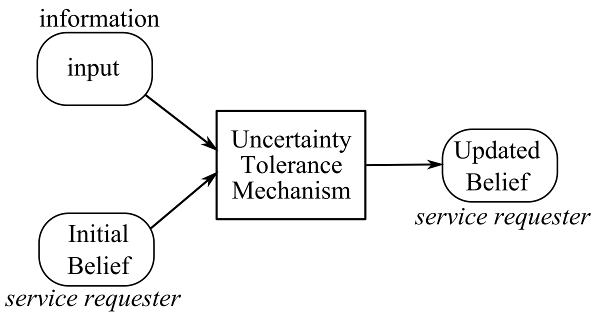


Figure 4.14: An Illustration of Overall Concept of Uncertainty Tolerance Mechanism

belief component refers to the service requester's belief towards the service provider regarding a specific situation or process in the service lifecycle. For example, upon receiving a service offer (in the pre-sp[1] phase) from a service provider, a service requester assigns an initial belief through the bootstrapping mechanism (as discussed in Section 4.5.4) towards the offer. Similarly, for the reporting of compliance from the service provider in the post-sp phase, the service requester can also assigns an initial belief towards the report. This initial belief is the uncertainty and classified as epistemic whereby the cause of it is due to the lack of information or knowledge about the situation.

The ***updated belief*** component refers to the service requester's initial belief that has undergone through the uncertainty tolerance mechanism and has an updated value. The updated value can either be the same, higher or lower that the initial value, depending on the input to the mechanism. The ***input*** component refers to a generic term given to any input that can assist the uncertainty mechanism in tolerating the uncertainty. Considering the epistemic nature of the uncertainty, in most cases the input refers to any information that can be used in the mechanism.

The ***uncertainty tolerance mechanism*** is the core component in the uncertainty tolerance concept whereby it refers to the process of utilizing the input in order to manipulate the initial belief and produces the updated belief. The exact implementation of this mechanism depends on the type of approach taken which will be presented in the next chapter.

[1]service provisioning

4.8 Summary

This chapter has presented three important contributions of the thesis which are (1) a unique classification scheme for uncertainty in service provisioning, (2) a unique view of uncertainty based on the temporal classification scheme and the service requester's perspective, and (3) the generic concept of uncertainty tolerance for service provisioning environment. Based on this concept, we will present a generic architectural framework for implementing uncertainty tolerance in a service-based system in the next chapter.

Chapter 5

An Approach to Uncertainty Tolerance in Service Provisioning Offer

5.1 Introduction

This chapter presents an approach to uncertainty tolerance. As a proof of concept, a specific uncertainty issue within the service provisioning lifecycle is chosen, which is the uncertainty towards the SLA offer in the pre-sp[1] phase. This particular issue is chosen since we believe it is a critical juncture for the service requester before actually committing to the intended service. The main contribution of this chapter is a unique approach towards uncertainty tolerance in service provisioning offer. To our knowledge, there is no existing work that attempts to provide an approach to tolerate uncertainty in service provisioning offer. The strength of this approach is that it includes well founded theory such as the subjective probability framework and decision making framework. This approach is of value to interested parties such as researchers, system designer, and system developer since it provides specific steps utilizing several theory in order to tolerate uncertainty in service provisioning offer. This chapter will also form the basis for the Uncertainty Tolerance Framework which will be presented in the next chapter.

The remainder of this chapter is organized as follows: Section 5.2 presents the underlying theory being utilized for uncertainty tolerance which includes discussion on

[1]service provisioning

probability framework, and evidence gathering. Section 5.3 continues with discussion on decision making under uncertainty which includes discussion on utility and expected utility theory. Section 5.5 summarizes the uncertainty tolerance concept.

5.2 The Underlying Concepts and Theory

The basic concept of our approach to uncertainty tolerance in service provisioning offer is to be able to represent service requester's subjective initial belief using a probabilistic framework (i.e. subjective probability) towards a proposition and then apply probability theory in the form of Bayesian Probability to manipulate this degree of belief. This section reviews the underlying concepts and theory that form our uncertainty tolerance approach which includes probabilistic framework, evidence gathering, and utility theory.

5.2.1 Probability Theory

To address the issue of uncertainty in service provisioning, we need to find a formal way of expressing the subjective customer's uncertainty towards the service offer (which is an uncertain event). Given an event, there are several ways to view the probability of the event occurrences:

1. **Classical view**: applies to equal probable event (for example, tossing a fair coin), whereby the probability of the event is the ratio between the number of outcome of favourable event and total number of outcomes. This view is conceptually simple for many situations but does not apply if the outcomes are not equally likely or when there are infinitely many outcomes.

2. **Frequentist view**: under this view, an event's probability is the proportion of times that the event will occur under large number of repeated trials. This view cover more cases compared to the classical view, but cannot be applied to event that hard or impossible to be repeated many times (for example, the number of success in launching rocket to the moon). Another challenge is to determine the number of trials to give good estimate of the limit.

3. **Subjectivist view**: a measure of an individual's belief in the occurrence of an event. Advantages of this view are: (1) it is applicable where the other views

cannot be applied (i.e. situations which require the inclusion of individual's belief), and (2) can vary between different individuals. On the other hand, this view requires coherence (consistency) condition in order to be workable. For example, is someone believe that there is a 70% chance of probability that its going to rain tommorrow, then to be coherent (i.e. consistent), the person cannot believe that the probability that its not going to rain tommorrow is 60%.

Out of the above three views, the subjectivist view is the right choice in addressing customer's uncertainty in service provisioning. One probability theory that fits well with subjective probability is Bayesian probability. In Bayesian Probability theory, a person's subjective belief (or formally termed as degree of belief) towards the possibility of an event or proposition, can be represented by a number between zero and one. The key to Bayesian Probability is Bayes' Theorem. In essence, Bayes' Theorem, which is a probability theory, shows the relationship between a conditional probability and its inverse[183]. The theorem is named after Reverand Thomas Bayes, who suggested the usage of the theorem to update beliefs. The theorem was published posthumously which was communicated to the to the Royal Society by Richard Price [184, 185].

The basis for Bayes' Theorem is conditional probability. Conditional probabilities are probabilities that rely on the value of another probability. The conditionality is represented by using a vertical slash "|", which can be read as ''given". The conditional probability of an event A, given an event B with $P(B) > 0$, is defined by

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{5.1}$$

The simple form of Bayes' Theorem can be expressed as follows:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{5.2}$$

whereby,

$P(A\|B)$	=	is the prior probability or marginal probability of A. It is "prior" in the sense that it does not take into account any information about B.
$P(B\|A)$	=	is the conditional probability of the B given A. It is also called the likelihood.
$P(A)$	=	is the prior probability of A. It is termed "prior" in the sense that it does not take into account any information about B.
$P(B)$	=	is the prior or marginal probability of B.

The numerator, $P(B)$ of the above Bayes' Theorem can be expanded using Total Probability Theorem which states,

$$\begin{aligned} P(B) &= P(A_1 \cap B) + \ldots + P(A_n \cap B) \\ &= P(A_1)P(B \mid A_1) + \ldots + P(A_n)P(B \mid A_n) \end{aligned} \tag{5.3}$$

where $A_1 \ldots A_n$ are disjoint events that form a partition of a sample space and assume $P(A_i) > 0$, for all i. Another way to represent Bayes' Theorem is using the conventional name of each term.

$$posterior = \frac{likelihood \times prior}{prob.of evidence}$$

5.2.1.1 How do we relate uncertainty tolerance with Bayesian Theory?

Once we have identified that Bayes' Theorem can be used to represent subjective probability, how do we relate this to the problem of uncertainty and uncertainty tolerance in service provisioning offer? Based on previous discussion, we have defined the customer's initial degree of belief towards an offer from a service provider as a state of uncertainty. This state of uncertainty is caused by lack of information or gap in knowledge about the future state of the event. For example, given an offer, the customer does not know how the event will turn out, i.e. in this case the SLA can either complies or fails.

To model the uncertainty in service provisioning offer, first we start with a set of *possible worlds* or *states*. These are the worlds or outcomes that the customer considered possible. For example, when receiving the SLA offer, it is reasonable to consider two possible worlds: (1) the SLA complies or (2) the SLA fails.

We can then define customer's degree of belief (i.e. uncertainty) as the *prior probability* based on Bayes' Theorem, given that this state of uncertainty is due to the lack (insufficient) of information. Through Bayes' Theorem, the prior (uncertainty) can be tolerated by obtaining evidence which eventually produced the posterior probability. Therefore, we consider the posterior probability as the *"tolerated uncertainty"*. An illustration of how Bayes' Theorem functions is illustrated in Figure 5.1.

We can now represent uncertainty tolerance using Bayes' Theorem through the followings steps. Firstly, we need to represent customer's uncertainty, i.e customer's initial

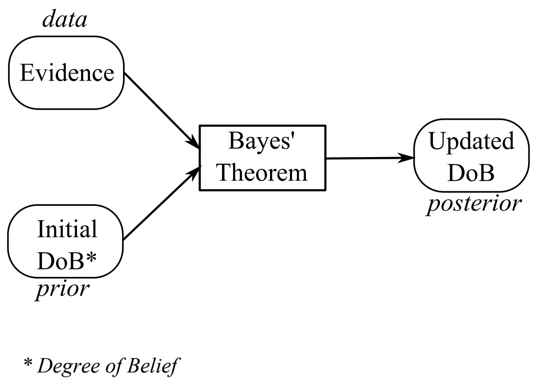


Figure 5.1: Representation of Bayes' Process

belief towards an SLA proposition. Lets define event S which represents whether an *"SLA comply"* and event $\neg S$ when *"SLA fail"*. The assumption is that the uncertainty is due to the lack of information about the future state of the event. Lets define this uncertainty as a *degree of belief*, represented by a probability function $P(S)$. This is also known as the *prior*.

Secondly, we need a way to present the *tolerated uncertainty*. The *tolerated uncertainty* is based on the conditional probability that the event S takes place given the evidence, e. Lets define *tolerated uncertainty*, which is the updated belief, as $P(S \mid e)$.

Finally, the collected evidence, given event S, which is also known as the likelihood, is given as $P(e \mid SLA)$ and the prior or marginal probability of the evidence as $P(e)$. Based on this information, the tolerated uncertainty can be presented as follows:

$$P(S \mid e) = \frac{P(e \mid S)P(S)}{P(e)} \tag{5.4}$$

Furthermore, using equation 5.3 to substitute $P(e)$, the above equation can be expanded into:

$$P(S \mid e) = \frac{P(e \mid S)P(S)}{P(e \mid S)P(S) \; + \; P(e \mid \neg S)P(\neg S)} \tag{5.5}$$

The next task is to justify the probability values for each of the probabilities in the above list. Firstly, how do we obtain the *prior*? Since the probability of prior is subjective, Bayes' theorem proponent allows an individual to place any arbitrary probability value (between zero and one) to indicate the individual belief towards the proposition. Furthermore, two different individuals can have different prior value towards the same proposition to indicate each individual's belief. The next section will provide a discussion on the justification of the value of prior, and Section 5.2.3 will provide discussion on the value of evidence.

5.2.2 Quantitative Value Assignment for Belief Bootstrapping

Previously, in Chapter 4, we have indicated that the service requester's initial degree of belief can be presented as a probability value between zero and one. This indicates that a service requester has to assign this value his/herself.

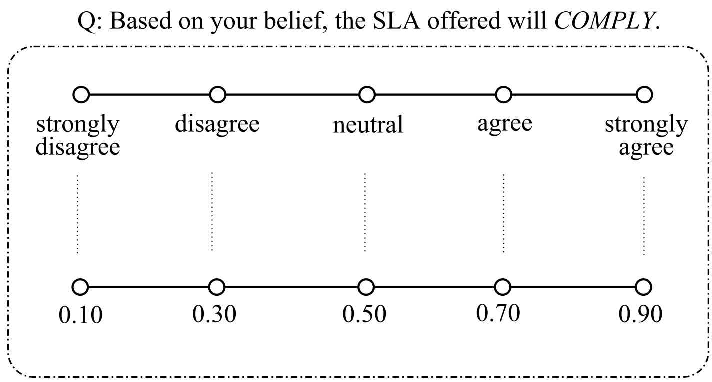


Figure 5.2: Likert-based Scale and Mapping to Assign Customer's Initial Belief

To simplify this process, we propose a qualitative Likert five point based scale between one to five, to indicate customer belief towards an SLA offer. Likert scale has been developed to measure attitude directly based on the response of a person towards a series of statements [186, 187]. This Likert-based scale is termed as *"Qualitative Belief Assignment"* (QBA) scale. In order to assign a quantitative value for the customer's initial degree of belief, the QBA is then mapped to a quantitative scale between 0.1 to 0.9 with a 0.2 increment to correlate with the five point QBA scale, as shown in Figure 5.2. The increment value of 0.2 is chosen to match the number of options (five) available in the Likert scale. This mapping is termed as *Quantitative Mapping* (QM). The finalized belief bootstrapping process is shown in Figure 5.3.

To summarize, the objective of the above discussion is to show how a customer can form a quantitative belief assignment towards service provisioning activities which are linked to uncertainty such as SLA offer from a service provider. Although under the subjective probability theory, an individual can assign any arbitrary value for the degree

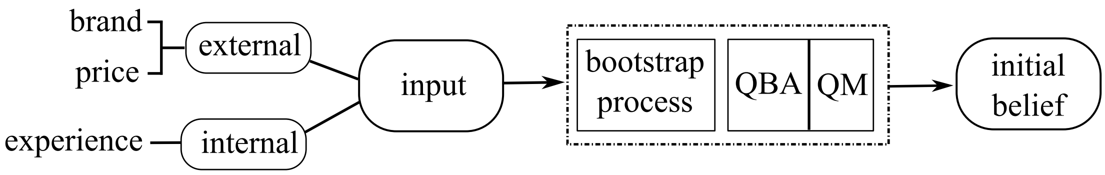


Figure 5.3: Initial Belief (prior) Bootstrapping Process with Different Type of Inputs, and QBA/QM modules

of belief (prior), we have provide justification on how to do so based on a systematic process.

5.2.3 Evidence

Evidence plays an important role in our uncertainty tolerance concept. Since the basic premise of the cause of uncertainty is the lack of information (epistemic), therefore, to tolerate uncertainty involves in getting additional information that can update customer's initial belief. The approach that we choose in tolerating uncertainty through Bayes' Theorem enables the inclusion of evidence in the process of uncertainty tolerance.

The next step is to determine suitable evidence source that is available within the context of service provisioning. One possible source of evidence is the interaction of other customers with the same service provider using similar service. The basic statement is *"If a number of customers consume similar service from the same service provider, what is the observation that when other customers Accept the offer and the service Comply?"*. This statement forms the basis of the evidence in our uncertainty tolerance mechanism.

5.2.3.1 Evidence Gathering

In other to gather this evidence, an evidence gathering mechanism framework is required, which will be discussed in the next chapter, under the overall Uncertainty Tolerance Framework. In essence, evidence gathering involves monitoring the activities of a set of other customers, recording some parameters to form a statistical evidence which will be used in the Bayes' Theorem. The evidence gathering is conducted in periodic mode whereby the activity is triggered when a customer receive an offer and initiate the uncertainty tolerance mechanism. One assumption for the evidence gathering activity is that at any period, there are several other customers within the ecosystem, i.e. interacting and consuming the service from the service provider.

There are two steps in evidence gathering: (1) collect multiple individual customer's interaction, recording specific parameters, and (2) fuse the multiple evidence sources into a single evidence value.

For the first step, we assume that each (other) customer interact with same service provider and consume similar service. We record the number of time each customer request a service and the outcome of each service invocation (comply or fail). Therefore, the individual interaction evidence, which is the probability for each customer can be expressed as follows:

$$e_i \ = \ p(e_i) \ = \ \frac{no.\ of\ S_{comply}}{N_S} \tag{5.6}$$

where e_i is the individual customer evidence, which is equivalent to the probability of $p(e_i)$. S_{comply} refers to the number of service invocation which complies to the SLA, and N_S refers to the total number of service consumption. In order to collect the most recent evidence for each customer, N_S can be implemented using *sliding window* mechanism whereby the value represents the most recent activities. Smaller value of N_S indicates the evidence collected is the most recent and should represents the most accurate information about the service compliance.

Secondly, once the individual evidence is collected, the next step is to fuse the different individual evidence by taking the average of the individual evidence, which can be expressed as follows.

$$E = \frac{\sum e_i}{N_e} \tag{5.7}$$

where E is the resulting fused evidence, $\sum e_i$ is the sum of the individual probability and N_e is the total number of individual customer involves in the evidence gathering activity.

5.3 Uncertainty Tolerance and Decision Making

In previous section, we have shown that it is possible to tolerate uncertainty (customer's initial belief) in service provisioning offer (by obtaining evidence). Unfortunately, the updated belief, in the form of probability value does not help the customer in decision making. Therefore, there is a need to provide a way for making decision based on the updated belief. One such mechanism is Expected Utility theory which is based on the concept of utility theory, and is widely used in economic fields.

5.3.1 Utility Theory

In order to apply the concept of Expected Utility Theory, we first need to understand the basic concept of utility theory. Utility theory has been used in economics to represent the measurement of satisfaction experiences by a customer when consuming goods or services. The principle of utility was first introduced by Jeremy Bentham in the beginning of 19th century [188] who suggested the measurement of quantities of pain and pleasure. The principle and theory of utility then become the topic of research by various economists such as Walras (demand function), Menger (marginal utility), Edgeworth, and Pareto (theory of indifference curve) [188].

There are two additional important facts about utility theory. Firstly, utility cannot be measured or observed directly, instead a relative utility is obtained by observing customer preference over a choice [189]. Secondly, there are two types of utility: (1) ordinal (ranking significant) and (2) cardinal (strength and ranking significant) utility [190, 191]. For the purpose of our research, we are employing ordinal utility since we are only interested on the ranking of decisions that a customer can take when deciding whether to accept or reject a service offer from a service provider.

In relation to uncertainty tolerance concept, we assume that the consumer has reasonable preferences about consumption in different circumstances. Lets recall the situation of uncertainty in which a customer has to decide about an offer (SLA) from a service provider. We have decided that there are two possible states that can occur, whether the SLA will *comply* or *fail*. We can denote the case where *"SLA comply"* as *state-1* and *"SLA fail"* as *state-2* with respective probabilities π_1 and π_2. Prior to the event, the customer does not know which states the event turn out to be, and after the event (post event), only one of the states will occur.

Next, we denote the customer's consumption if state-1 occurs as c_1 and if state-2 occurs as c_2. The customer, prior to the event occurring must select between the various bundles (c_1, c_2). Post-event, the customer will get either one of the c_1 or c_2 depending on which state has occurred. Since the two states are mutually exclusive (i.e. only one of them can occur), then $\pi_2 = 1 - \pi_1$. Given this notation, the utility function for consumption in state-1 and state-2 is as follows:

$$u(c_1, c_2, \pi_1, \pi_2) \tag{5.8}$$

5.3.2 Constructing a Service Requester's Utility Function

There are several approach in eliciting or constructing utility of outcomes of choice in a probabilistic situation for an individual service requester. Existing approaches are (1) direct elicitation of utility (using von-Neumann approach, modified von-Neumann, or Ramsey method) [192] (2) risk interval approach [193], (3) experimental methods (using real financial rewards through gaming method) [194], and (4) observed behaviour (relationship between actual behaviour and empirically specified models are compared) [195].

In our uncertainty tolerance approach we does not include the utility eliciting mechanism since we assume that a service requester has already obtain those utility values using one of the possible approaches.

5.3.3 Expected Utility Model

"Expected Utility" (EU) model is a well known concept which is widely used in economics to solve the issue of choice under uncertainty. Expected utility, can also be referred as *"probability-weighted utility theory"*, due to the fact that the EU for each alternative is the weighted average of its utility values under different states, whereby the probability is used as the weight.

We are considering a situation of uncertainty in which a service requester does not know before the event which states of the event will occur. However, the service requester can list the various possibilities and can assign probabilities to them. For simplicity, here we assume two possible states of the world, *state 1* (refers to SLA comply), and *state 2* (SLA fail), with respective probability π_1 and π_2. The term *world* refers to the system or environment where the event take place (for example service provisioning). Prior to the event, the service requester does not know which of these states will occur. After the event, one and only one of the states will occur.

We represent c_1 to the individual service requester's consumption if *state 1* occurs and c_2 the individual service requester's consumption if *state 2* occurs. The service requester must choose prior to the event between the various uncertain (risky) bundles (c_1, c_2). After the event the service requester will get one of c_1 or c_2 depending upon which state of the world has occurred. Based on the utility function in 5.8, the expected utility function can be expressed as follows:

$$U(c_1, c_2, \pi_1, \pi_2) \;=\; \pi_1 u(c_1) + \pi_2 u(c_2) \tag{5.9}$$

whereby,

$u(c_1)$	=	utility from individual service requester's consumption if *state 1* occurs.
$u(c_1)$	=	utility from individual service requester's consumption if *state 2* occurs.
π_1	=	probability for *state 1* to occur.
π_2	=	probability for *state 2* to occur.

In the above equation 5.9, the probabilities are given by the problem so the only element that needs to be specified is the function $u(.)$. This function is known as the von Neumann-Morgenstern (vNM) utility function and was introduced by mathematician John von Neumann and economist Oskar Morgenstern in a book titled, *"The Theory of Games and Economic Behavior"* [196]. Given the function $u(.)$, which tells how much utility is obtained from some amount of consumption, the explanation for equation 5.9 is as follows: with probability π_1 *state 1* happens and the service requester consumes c_1 from which he or she gets utility $u(c_1)$; with probability π_2 *state 2* happens and the service requester consumes c_2 from which he or she gets $u(c_2)$. The right hand side of equation 5.9 is the utility that the service requester *expects* to get from the "before the event" uncertain (risky) bundle (c_1, c_2).

Expected utility theory is the foundation in decision making under uncertainty [197, 198, 199] and also an important aspect in modern game theory [196]. In general, the *vNM* Expected Utility function can be mathematically expressed as:

$$EU(d) \;=\; \sum_{s \subset S} P(s)U(s) \tag{5.10}$$

where $EU(d)$ is the expected utility of decision d, which a subset of possible actions $\{Accept, Reject\}$. $P(s)$ is the probability of outcome s, and $U(s)$ is the utility of outcome s, whereby s is a subset of possible outcomes $\{comply, fail\}$. Therefore, given a proposition, *SLA-status* and possible actions, $d = \{Accept, Reject\}$ with respective utilities, we can calculate the Expected Utility for each of the actions as $EU(Accept)$ and $EU(Reject)$. The maxim of *"Maximizing Expected Utility"* (MEU) is to select the higher value EU out of the possible outcomes [200]. Therefore, by comparing $EU(Accept)$ and $EU(Reject)$, and selecting the higher value of the two, we have helped customer's in decision making under uncertainty.

5.4 Cost Consideration for Uncertainty Tolerance

In addition to the cost associated with executing a service [201, 202], the approach towards uncertainty tolerance presented in this chapter does incurs additional cost. To simplify our approach and subsequent framework and empirical study, the cost factor is not considered in this thesis but will be briefly discussed in this section to highlight this issue to the reader. In general, the term **cost** refers to any factor related to the uncertainty tolerance approach that poses negative consequence to the entities involved, in particular to the service requester (i.e. customer). Possible cost associated with the approach can be divided into several types as follows:

- **Time Cost:** *"time cost"* is referred to the additional time taken in order to gather evidence from N number of other service requester. In the above approach, the evidence gathering process is triggered when the service requester initiate the uncertainty tolerance mechanism upon receiving the service offer from the service provider and terminates when sufficient evidence has been collected (in this case, for N number of service requesters). Therefore, depending on the amount of time taken, this waiting period (if its too long) can be of negative consequence to both the service requester and provider.

- **Resources Cost:** *"resources cost"* is the overhead incurred mainly to the service provider in term of additional resources that needs to be deployed to implement the uncertainty tolerance mechanism especially the evidence gathering process. This resources can be in the form of additional hardware, software, and/or man-power.

- **Complexity Cost:** the introduction of uncertainty tolerance mechanism into the service provisioning system inadvertently resulting in added complexity to the overall process of service provisioning. The relationship and interaction between multitude of different modules and processes in the uncertainty tolerance mechanism give rise to the complexity. The added complexity does not directly correlate to an increase of cost, but can result in an increase in time taken (i.e. time cost) and/or resources (i.e. resources cost).

For the entities involved in a service provisioning transaction, in particular the service requester, the benefit of the uncertainty tolerance mechanism need to be balanced with the cost incurred in implementing such mechanism. We acknowledge the existence of the above cost and the potential counter effect to the benefit gained, but to limit the scope of the thesis, the quantitative relationship or impact of cost is not investigated in this thesis. This issue will be highlighted as a potential thread of future research work in Chapter 8.

5.5 An Approach to Uncertainty Tolerance: The Overall Picture

In this section, we will summarize the overall picture of the Uncertainty Tolerance concept, compiling the various theory and concept discussed in previous sections. Figure 5.4 illustrates the overall concept of uncertainty tolerance with respect to the uncertainty problem in service provisioning offer. There are three components: (1) Bayes Inference, (2) Evidence Gathering, (3) Expected Utility Model.

- **Bayes Inference (BI)**: This component is the key component in uncertainty tolerance concept. Its main objective is to update customer's initial belief (which we classify as epistemic uncertainty) by reducing the gap in knowledge.

- **Evidence Gathering (EG)**: This component functions as a support component to the Bayes Inference component. The main objective of EG is to gather evidence, and perform calculation.

- **Expected Utility (EU)**: This component enable a customer's to make decision based on the updated belief.

The above three components form the basis of the uncertainty tolerance approach towards service provisioning offer uncertainty.

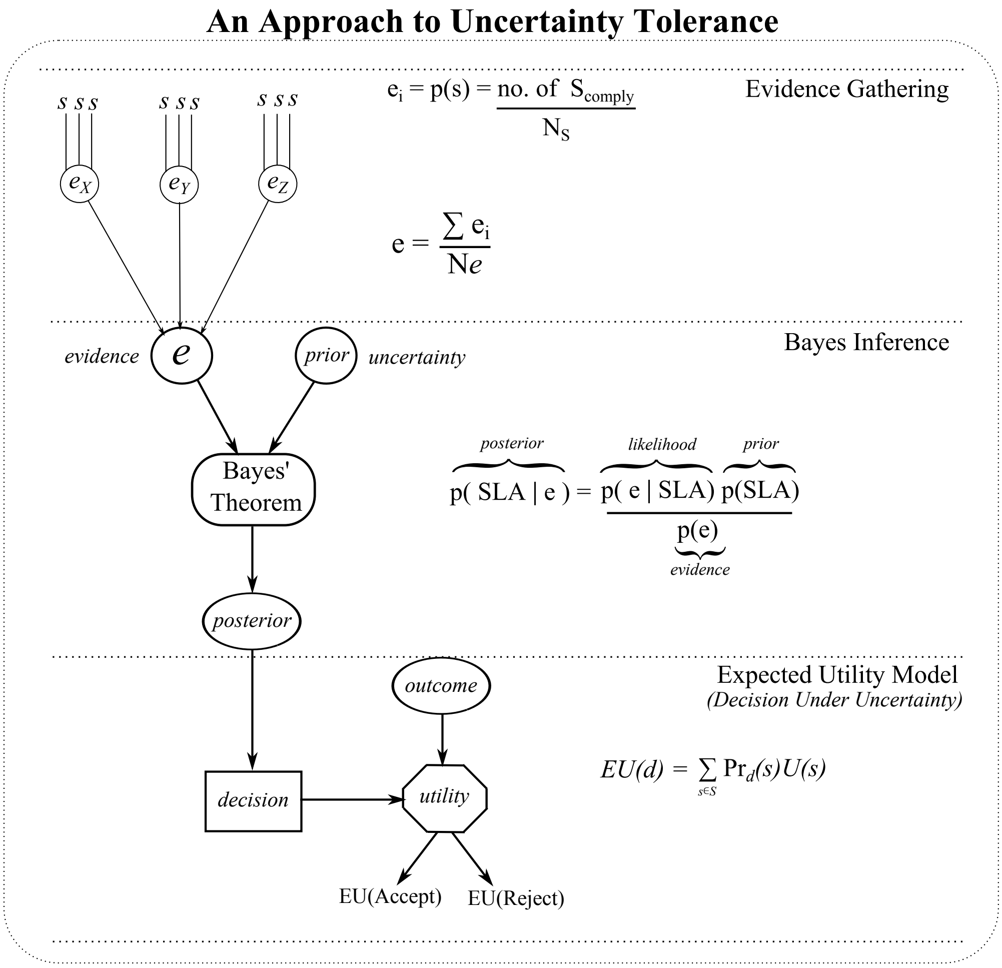


Figure 5.4: An Approach to Uncertainty Tolerance in Service Provisioning Offer

5.6 Summary

This chapter presented the approach taken to tolerate the uncertainty in the service provisioning offer, through the application of several theory and concept such as the subjective probability, utility, and evidence. To our knowledge, such approach is unique within the context of service provisioning and has not been attempted by other researchers. Based on this approach, we will present an architectural framework for implementing uncertainty tolerance in a service-based system in the next chapter.

Chapter 6

A Framework for Uncertainty Tolerance in Service Provisioning Offer

6.1 Introduction

This chapter addresses the problem of uncertainty in service provisioning offer from the context of the service requester (i.e. customer), utilizing the general concept of uncertainty tolerance presented in Chapter 4 and the approach presented in Chapter 5. The main contribution of this chapter is a concrete framework which provides a generic implementation of uncertainty tolerance mechanism. To our knowledge, no such attempt has been made to design and develop such framework to address uncertainty in service provisioning offer. The value of such framework is to provide an implementation blueprint for system designer and system developer as a guideline to design and develop a service-based system with uncertainty tolerance capability.

The remainder of this chapter is organized as follows: Section 6.2 lays down the requirements and assumptions of the framework. Section 6.3 introduces important terms used throughout this chapter with regards to the framework, and introduces the generic architecture for the framework. The next section, Section 6.5 presents the overall picture of the framework. Section 6.5 presents the key contribution in which the different sub-components of the uncertainty tolerance engine are described in details. Section 6.6 provides the overall sequence diagram of interactions between the entities.

Finally, Section 6.7 summarizes the chapter highlighting key finding and contributions.

6.2 Framework Requirements and Assumptions

The uncertainty tolerance framework presented in this chapter is based on the following key assumptions:

1. Rational agent: The service requester (i.e. customer) is considered as a rational agent and is risk neutral. Under *Bayesianism*, a rational agent chooses the option with the highest expected utility.

2. Trusted Service Provider and Evidence source: The framework operates under two trust assumptions: First, the SLA compliance status reported by the service provider after each service consumption is trusted, i.e. the service provider does not lie about the status of the SLA compliance. Secondly, the evidence collected from multiple sources is also trusted, i.e. the sources do not lie or untruthfully submit the evidence.

6.3 Framework Fundamentals

The term ***framework*** used in this chapter and the thesis refers to an application level development blueprint which provides two important information. Firstly, the framework shows components required for the uncertainty tolerance mechanism to function, plus the logic behind each of the sub-components. Secondly, it provides the logical flow of the uncertainty tolerance process based on the service provisioning lifecycle.

Beside the term *framework*, there are several other terms used throughout this chapter that need explanation and definition. An ***entity*** in the framework refers to the parties involve in a service provisioning lifecycle, namely the *service requester*, *service provider*, and *trusted third party*. Each entity contains an agent and each entity's action is conducted through the agent. In the context of the framework, an ***agent*** is defined as a computer program that acts on behalf of an entity. This definition is inline with the definition given by the Web Services Architecture Working Group Note [83]. An ***engine*** in the framework is a conceptual term that implies a set of processes to achieve an outcome. An engine consists of one or more building blocks which is termed as component. A ***component*** is a logical grouping of one or more associated functions. A ***function***, is defined as a discrete block of computational logic to perform a task and/or achieve a certain result. This is similar to the notion of *procedure*, or *method* used in programming language [203, 204]. A *function* is the basic building block in the framework.

There are three types of function: (1) ***framework functions***, which is simply referred as *function*, are functions that implement key functionality of the uncertainty tolerance concept, (2) ***auxiliary functions*** are functions that are functioning in a supporting capacity role. In essence, we are only interested in the value that the function returns or the logical process that the function implements to arrive at a certain state, thus we do not implement the *auxiliary functions*, and (3) ***system function*** refers to the functions that implement the underlying processes within a service provisioning lifecycle.

6.3.1 Generic Architecture of the System

A ***system*** is defined as a group of entities that interact with each other. In the context of service provisioning, we have defined three type of entities that make up the system, which are service requester, service provider, and service directory. In following section, the service directory entity will be known as the Trusted Third Party (TTP), that combine both the service directory functionality plus the uncertainty tolerance mechanism. Figure 6.1 illustrates the system with the aforementioned entities.

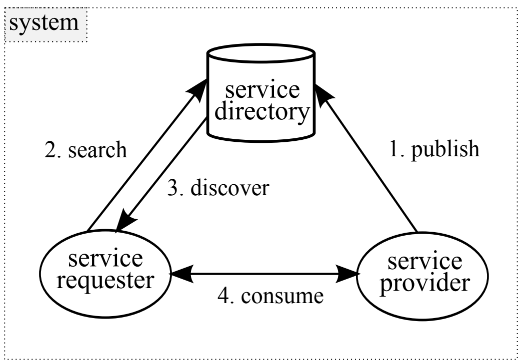


Figure 6.1: Generic Structure of a System

A ***relationship*** in the *system* is defined as a connection between two entities in the system which is represented by an arc connected between those two entities, as shown in Figure 6.1. The connection represents the action or process between the entities. Depending on the required interaction, a relationship can be a single or two way interaction between entities.

6.3.2 Placement Consideration for Uncertainty Tolerance Engine

Another important consideration is for the placement of the uncertainty tolerance engine. In general, there are three possible locations for the engine, based on the *entity* in the system, which are: (1) at Trusted Third Party (TTP), (2) at Service Requester (SR), and (3) at Service Provider (SP). Rana et. al. [109] discusses three possible locations with respect to monitoring as follows:

- **Trusted Third Party:** the TTP is an independent entity that can monitor and log activities between the service requester and service provider location. The key requirement is that the TTP has to be trusted by both *SR* and *SP*. To provide non-repudiation and reputation for both *SR* and *SP*, a signed ticket is generated after the service has been completed and send to both parties. One drawback of TTP location is that it is not possible to monitor activities internal to either the *SP* or *SR*.

- **Service Provider:** The second option is to implement the monitoring module at service provider location which has equivalent functionality as TTP, i.e. able to monitor and log activities between itself and service requester, but with the advantage of monitoring the internal activities or state of the service provider. However, there are two drawbacks: (1) *SP* might not revealed or report the full information about its internal state and selectively choose information which is beneficial to itself, and (2) *SP* might falsely report the actual outcome of the monitoring (i.e. whether violation occurs for SLOs).

- **Service Requester:** The third option is to implement the monitoring module at the service requester location. In term of functionality, it is equivalent to the TTP, in which the service requester needs to determine if a SLO[1] violates the agreement. Unfortunately, to prove such violation to the TTP and SP is difficult. For example, in monitoring delay SLO, the service requester is unable to

[1]Service Level Objective

distinguish between network delay (which is not under *SP* controls) and processing delay in *SP* location. Therefore, the value for implementation of monitoring module at service requester location is to provide a means to establish a measure of trust towards the service provider.

Based the above discussion, in order to achieve highest trust confidence for both the service requester and service provider, the most logical placement of the uncertain tolerance engine is an entity that does not have malicious intent or can benefits from cheating or abnormal behaviour. Thus, the **Trusted Third Party**, which is deemed trusted by the other two entities in a service provisioning system is chosen as the location for implementing the uncertainty tolerance engine. Trusted Third Party (TTP) is a common approach to facilitates interaction between two or more parties in many areas such as privacy in multi parties transaction [205], online retailing [206], electronic commerce [207], and telemedicine [208].

6.4 General View of the Framework

Figure 6.2 shows the general view of the framework for uncertainty tolerance. The framework consists of various components located in different entities. The key component is the uncertainty tolerance engine located in the Trusted Third Party entity, which perform the underlying belief updating essential in our choice of implementation for the uncertainty tolerance mechanism. Each of the components will be discussed in detail in the following sections.

One property of the framework is that it should be as generic as possible so that it is possible to implement the framework in any service provisioning system. For example, the *validation logic* module discussed in Section 6.5.3 uses the *execution time* as the metric being validated by the module, but the *validation logic* module can also cater for other metrics such as *delay*, and *uptime*. One limitation is that, in its current implementation, the module (and subsequently the framework) is unable to cater for composite metric (combination of several metrics), and can be addressed in future works.

Several processes such as the *register*, *search*, *consume*, and *offer* are not discussed from the implementation point of view since those processes are part of the underlying specific service provisioning system implementation, such as web services [83].

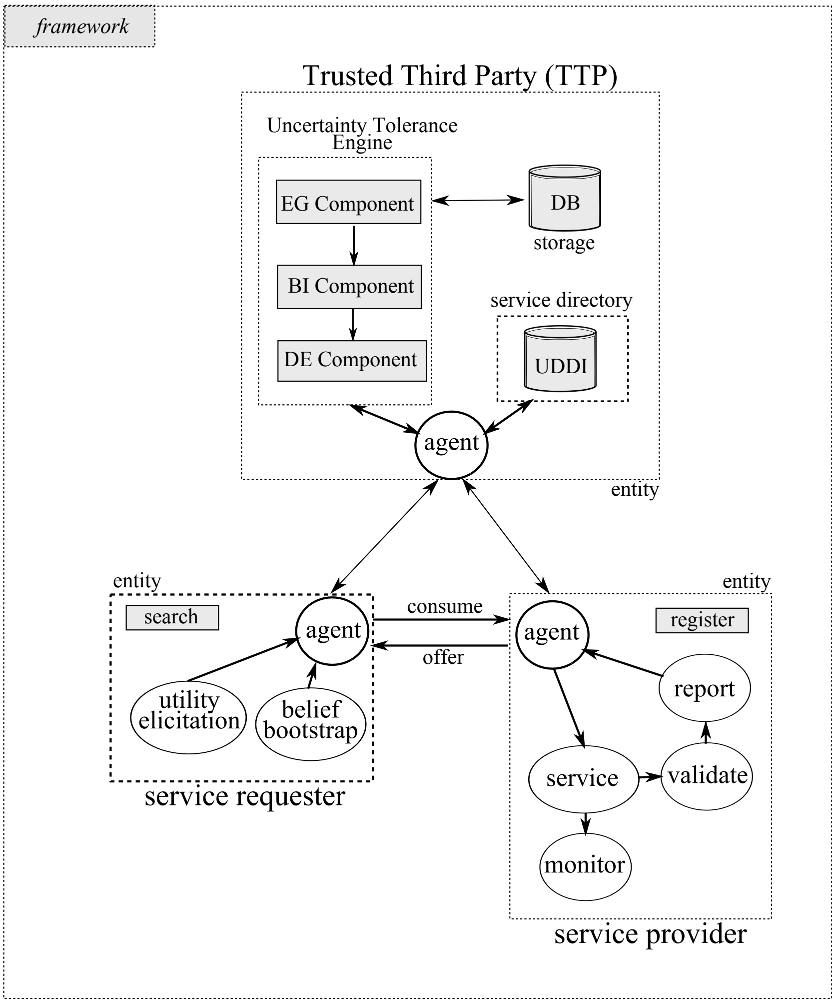


Figure 6.2: General View of the Uncertainty Tolerance Framework

6.5 Framework Components

This section provides detail discussion on each of the components and functions within the framework, focusing on the logic of each of the components. The breakdown of the following discussion is based on entities in the framework.

6.5.1 Trusted Third Party (TTP) Entity

TTP is one of the entity in the system and contains the key element of the uncertainty tolerance framework, which is the *Uncertainty Tolerance Engine* (UTE). The TTP entity serves two main functions: (1) as a *service directory*, and (2) host for the

Uncertainty Tolerance Engine (UTE). Data required for the UTE operation is stored in a chosen storage system such as a simple flat text file (etc. CSV[1]) or a full fledge relational database such as MySQL[2] or a self-contained, serverless database such as SQLite[3]. Any interaction with other entities in the system is conducted through the agent.

The UTE consists of three main components: (1) Evidence Gathering (EG) component, (2) Bayes Inference (BI) component, and (3) Decision Engine (DE) component.

6.5.1.1 Evidence Gathering Component

Evidence, in the context of uncertainty tolerance is defined as any information from the service provisioning process that can reduce the gap in knowledge of a customer when presented with an SLA offer. Evidence can be in the form of statistical data collection or opinion of an expert about a particular issue. Figure 6.3 shows the Evidence Gathering component (EG-Comp) with sub-components. The main functionalities of EG-Comp are to collect service record for a set of customer and perform calculation to generate the evidence value.

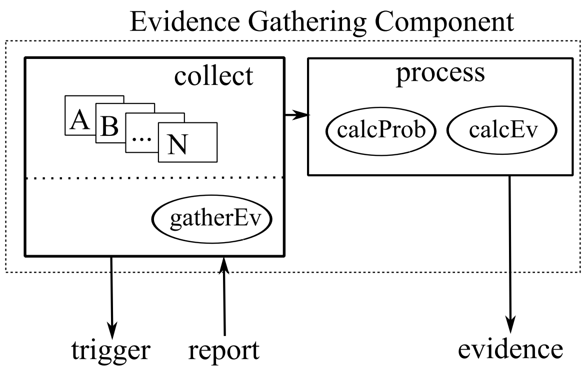


Figure 6.3: Evidence Gathering Module

The EG-Comp consists of two main sub-components: (1) collect component (collComp) , and (2) process (procComp). The *collect component* acts as a record gatherer for a set of customers who are using the same service from the same service provider. When the EG module is triggered, the collComp will query the service provider to get

[1]Comma Separated Value
[2]http://www.mysql.com
[3]http://www.sqlite.org

a list of customers who are either in the process or about to start service interaction. For each customer in the list, the following GatEv function 1 will be executed:

Function 1 Gather Evidence

```
function GatEv(runNum)                          ▷ runNum is set to an integer value
    countService ← 0                            ▷ no. of service consumed
    countComply ← 0                             ▷ no. of service comply
    for i ← 1, runNum do
        countService := countService + 1
        trigger ConsumeService()
        receive Report(comply)
        if comply == TRUE then
            countComply := countComply + 1
        end if
    end for
end function
```

In essence, the GatEv function maintains a record of how many services have been consumed and how many of those services complied with the SLA.

The *process component* is triggered once the collComp components has stopped. The main objective of procComp is to (1) calculate the probability of each customer interaction, and (2) combine probabilities of all customers involve in the evidence gathering process.

Function 2 Calculate Probability

```
function calcProb(servComp, servTotal)
    e ← 0                ▷ evidence value (i.e. probability of each customer)
    if servComp != 0 AND servTotal != 0 then
        e = servComp/servTotal
    end if
    return e                                    ▷ return calculated evidence
end function
```

The calcProb function as shown in function 2, calculates the probability of compliance for a set of service consumption by a customer. This function is called after the evidence gathering process for each customer has stopped. The return value, e is the probability value which will be utilized in the evidence fusion process.

Function 3 Calculate Combined Evidence

```
function calcCombEv(listOfEv, NoOfCust)
    sumEv ← 0                                  ▷ sum of evidence
    evComb ← 0                                 ▷ combined evidence
    for i ← 0, NoOfCust do
        sumEv := sumEv + listOfEv[i]
    end for
    evComb := sumEv/NoOfCust
    return evComb                              ▷ return combined evidence
end function
```

Once the evidence gathering process for each customer has stopped, and the probability value has been calculated using function 2, the *Calculate Combined Evidence* function as shown in function 3 will be executed. This function combine the individual evidence value for each customer in a given set of customers who are using the same service from the same service provider into a single value, termed evidence, noted as *evComb*. The *evComb* value will be used in the Bayes' Inference Component.

6.5.1.2 Bayes' Inference Component

The Bayes' Inference Component (BI-Comp) is a key component in the UTF. Its main function is to update customer's initial belief (the prior) based on gathered evidence, as shown in Figure 6.4. The inference component relies on Bayes' Theorem as its operational foundation.

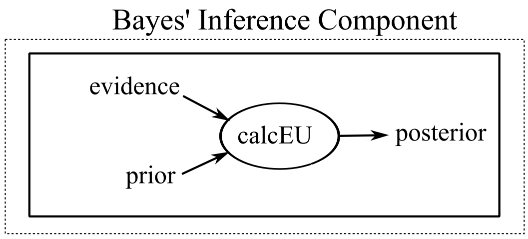


Figure 6.4: Bayes' Inference Module

The calcPost function, as shown in function 4 is the only function in the BI-Comp, and calculate the updated belief (i.e. posterior value) for a customer based on Bayes' Thoerem. The function requires two inputs, the prior value and combined evidence value calculated in function 3, and return the calculated posterior value, *post*.

Function 4 Calculate Posterior

function CALCPOST($evComb, prior$)
    $post \leftarrow 0$ ▷ tolerated uncertainty, i.e. posterior
    $num \leftarrow 0$ ▷ numerator portion of Bayes' Thorem
    $denom \leftarrow 0$ ▷ denominator portion of Bayes' Thorem
    $num := evComb * prior$
    $denom := num + (1 - evComb) * (1 - prior)$
    $post := num/denom$
    return *post* ▷ return calculated posterior
end function

6.5.1.3 Decision Engine Component

The Decision Component (DE-Comp) functions as a decision making tool for a customer under uncertain situation by calculating the expected utility value of each of possible action taken by a customer. the DE-Comp consists of two main functions: (1) calcEU function, and (2) evalDec function, as shown in Figure 6.5.

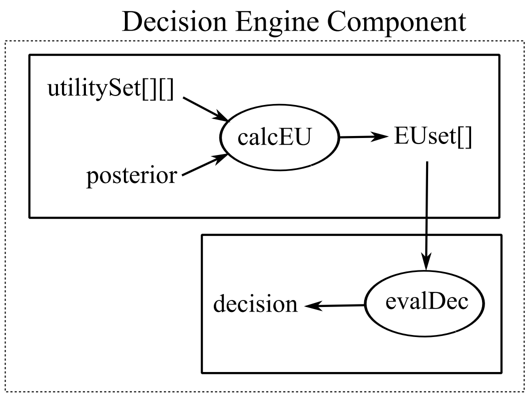


Figure 6.5: Decision Engine Module

The caclEU function, as shown in function 5 calculates the expected utility value for each possible decision in the decision set by a customer. The function requires two input: (1) a set of utility values for the decision set, and (2) the posterior value calculated in function 4. The function returns a set of expected utility values which will be used in the evalDec function.

The evalDec function, as shown in function 6, evaluate the expected utility values calculated by function 5. The return value of the function indicates which action should

Function 5 Calculate Expected Utility Value

```
function calcEU(utility[ ][ ], post)                ▷ utility[ ][ ] set of utility values
    EUSet[ ] ← 0                          ▷ Expected Utility array for {Accept,Reject}
    EUSet[Accept] :=post*utility[0][0] + ((1-post)*utility[0][1])
    EUSet[Reject] :=post*utility[1][0] + ((1-post)*utility[1][1])
    return EUSet[ ]                              ▷ return set of expected utility value
end function
```

be taken by a customer. For example, when the expected utility value for the decision *Accept* is greater or equal to the decision *Reject*, the function will return boolean value *true*, or otherwise.

Function 6 Evaluate Decision

```
function evalDec(EUSet[ ])                 ▷ EUSet[ ] set of expected utility values
    if EUSet[Accept] ≥ EUSet[Reject] then
        return true
    else
        return false
    end if
end function
```

6.5.2 Service Requester Entity

Service requester is a generic term for an entity that require a service to complete a certain task or achieve a specific objective. In other word, the term is synonym with the terms *"customer"*, *"end user"*, or just *"user"*, which will be used interchangeably throughout this chapter. As shown in Figure 6.6, there is one system function, *search*, used by the service requester to search for required service from the service directory in the TTP, and there are two auxiliary functions, (1) utility elicitation, and (2) belief bootstrap. The utility elicitation function provides the utility value for the service requester with regards to the possible actions that can be taken on the (SLA) offer provided by the service provider. Implementation of this function is not provided and we assume that for any given service offer, the utility value has already been elicited to be used in other function or component. The utility elicitation function can employ any of the following utility elicitation method such as (1) direct scaling method [209], (2) certainty-equivalent method [210], and (3) probability-equivalent method [210]. The utility value is passed to the TTP and stored until retrieved for usage.

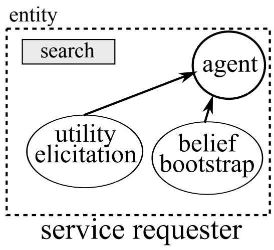


Figure 6.6: Service Requester Entity

The belief bootstrap function facilitates the process of initializing the qualitative value of the service requester's initial degree of belief, i.e. the prior. The justification for the process has been discussed in Chapter 4. The resulting prior is passed to the TTP for storage until retrieved for usage. All communication from these two functions is conducted through the agent.

6.5.3 Service Provider Entity

A ***Service provider*** is an entity that provides communications service, storage service, or processing service or any combination of these services to other entities. As shown in Figure 6.7, there is one system service, *register* which is used by the service provider entity to publish its services into the *service directory* in the TTP, one auxiliary function, *service*, and three functions: (1) monitor function, (2) validate function, and (3) report function.

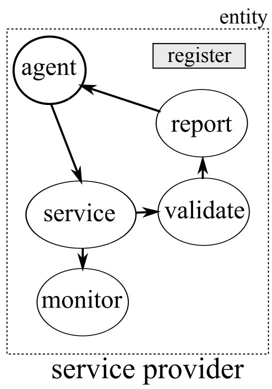


Figure 6.7: Service Provider Entity

The *service function* represents the core process that accomplish the required computation and processing that define the service offered to the service requester. The monitor function does the monitoring and collecting of specific metrics related to the service such as *processing time*. Once the service has completed, the monitor function returns the collected metric data to the *service function* represents the core process that accomplish the required computation and processing that define the service offered to the service requester. The monitor function does the monitoring and collecting of specific metrics related to the service such as *processing time*. Once the service has completed, the monitor function returns the collected metric data to the validate function. The *validate function* evaluates the collected metric and determine whether the SLA has been fulfilled. The *validate function* then return the status of the validation process (a boolean value).

The validate function contains a simple violation validation logic which compares the collected/monitored metric against the agreed threshold of the metric stipulated in the SLA. For example, *execution time* is defined as the time taken to complete the required task (measured on service provider side). Lets define *execution time* as T_{ET} and set the threshold requirement for this Service Level Objective using a *LogicalOperator* and *PropertyValue*. For example, an SLO statement such as *'the execution time of the service must be less than 500ms for each service execution"* can be represented as $T_{ET} < 500ms$. Therefore, the validation function includes validation logic that compares the monitored value of T_{ET} against the threshold and returns a boolean status. Figure 6.8 illustrates the SLO validation logic process.

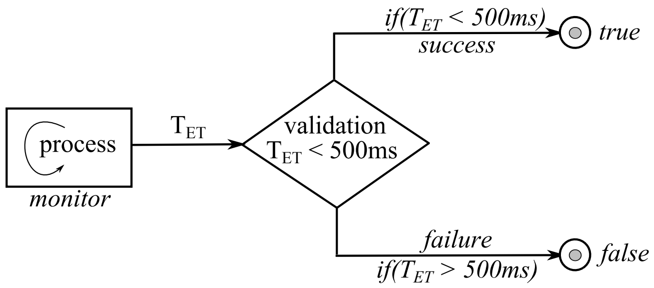


Figure 6.8: SLO Violation Validation Logic

The *report function* collate the status of the validation from the *validate function* and the metric collected from the *monitor function* into a *report*. A *report* is simply a textual representation of the collated values which can be stored to local storage or send to the *service requester* after the service has terminated.

6.6 General Sequence of Interactions

Figure 6.9 shows the overall sequence diagram for interactions between the three entities in-line with the lifecycle of the service provisioning discussed in previous chapters. The three entities shown in Figure 6.9 are (1) service requester (i.e. the customer), (2) service provider, and (3) Trusted Third Party (TTP) (which also includes the service directory).

This sequence diagram will be useful to interested parties to understand the steps taken by various entities in the service provisioning lifecycle in order to achieve uncertainty tolerance for the uncertainty in service provisioning offer. Steps 1 to 3 shown in Figure 6.9 refer to the basic interactions between the three entities which include the register, search, and discover service. These steps are part of the original interactions in a service lifecycle, thus will not be discussed in detail. Once a service requester has discovered the required service, the service requester will send a request offer to the service provider in step 4. The service provider will respond to the request by replying with an offer back to the service provider in step 5.

After receiving the offer, service requester will initiate step 6 (utility elicitation) and step 7 (belief bootstrapping). Step 6 will generate utility values for the service requester and step 7 will generate the initial degree of belief (i.e. the uncertainty) for the service requester. These data which will be stored and retrieved at later stages. Once steps 6 and 7 have completed and the values/data have been stored, service requester will initiate the uncertainty tolerance mechanism by sending a request to the TTP in step 8. At TTP, step 9 (gather evidence) will be executed, followed by step 10 (Bayes' inference), and step 11 (decision engine). Step 11 involves the calculation of expected utility value and decision logic that evaluates the expected utility values. The status of the evaluation (boolean) is then returned back to the service requester in step 12.

Th service requester then can use the returned status *status* to make decision, i.e. if the status returned us *true*, then the service requester will *accept* the offer as shown in step 13. The service requester then consumes the service in step 14 while the service requester monitors the service invocation in step 15. Once the service operation is

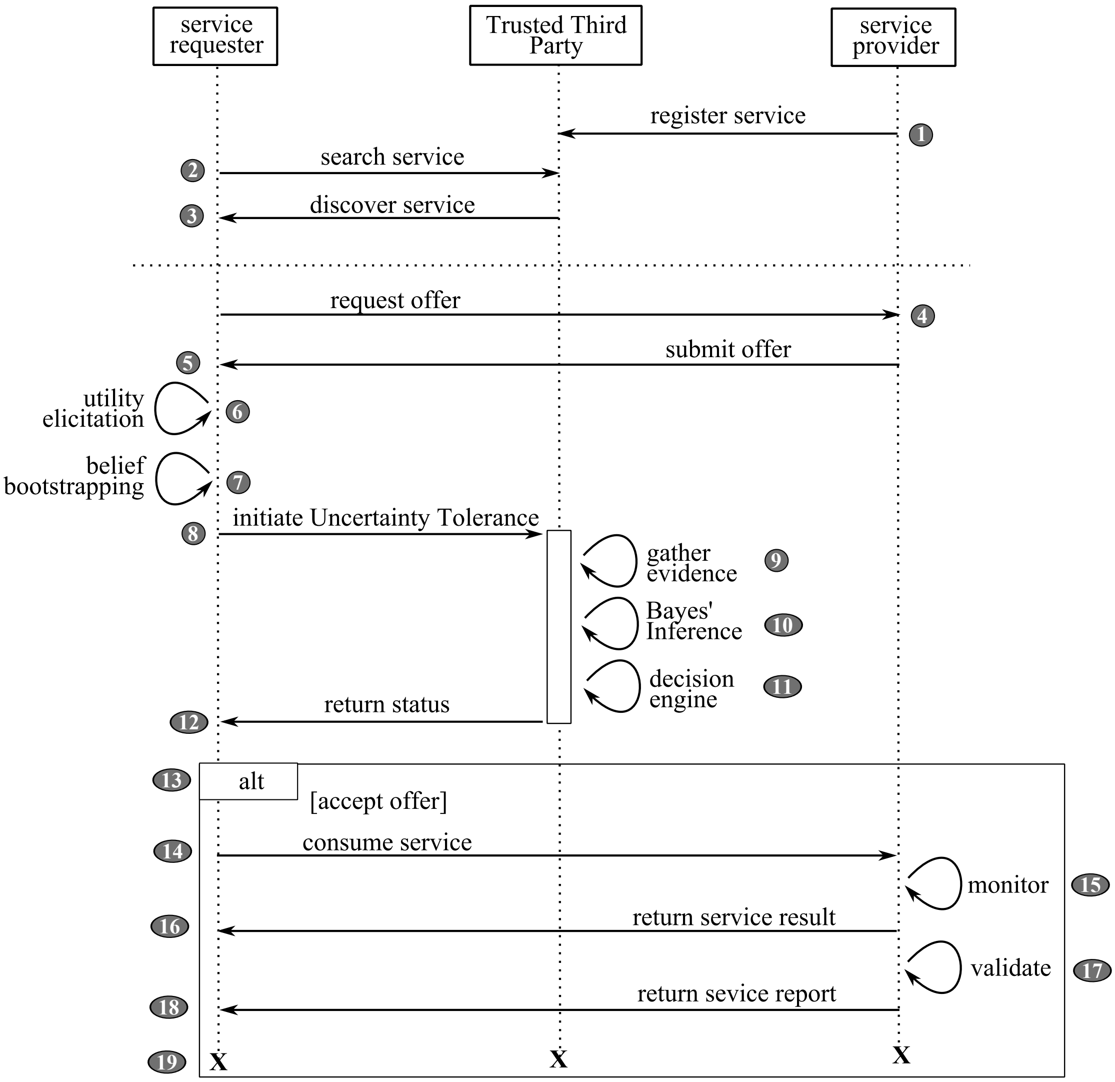


Figure 6.9: Overall Sequence Diagram for Entities Interactions

completed, the result (if any) is returned to the service requester as shown in step 16. In step 17, the service requester validates the collected data in step 15 and send a service report to the service requester in step 17. The lifecycle and interaction among the entities terminate in step 19.

6.7 Summary

This chapter presents a generic framework on how the uncertainty tolerance approach discussed in Chapter 5 can be implemented. The framework presented in this chapter includes details explanation of the main functions involved in the uncertainty tolerance process. This framework will be useful to system designer and developer who are involved the design and development of service provisioning system and would like to include aspect of uncertainty tolerance in the system. This framework is unique due to the fact that this is the first attempt to provide such framework that deals with the problem of uncertainty tolerance in service provisioning offer, which functions as a blueprint to implement uncertainty tolerance measure. The following chapter presents an empirical study based on simulation design using the uncertainty tolerance framework presented in this chapter.

Chapter 7

An Empirical Study of Uncertainty Tolerance in Service Provisioning Offer

7.1 Introduction

This chapter presents an empirical study to validate the viability of the uncertainty tolerance concept presented in Chapter 4, the approach towards uncertainty tolerance presented in Chapter 5, and to evaluate the uncertainty framework presented in Chapter 6. The outcome of the study would be valuable to interested parties, in order to understand how the uncertainty tolerance mechanism affect service requester's belief and also demonstrates the positive outcome of the mechanism in dealing with uncertainty in service provisioning offer.

The remainder of the chapter is organized as follows: Section 7.2 provides a scenario as the motivating example for the empirical study. Section 7.3 discusses the requirements that the empirical study should fulfill. In short, the requirements reflect the outcome of the uncertainty tolerance concept discussed in Chapter 4 and the specific approach presented in Chapter 5. Section 7.4 lists assumptions being made for the study. Section 7.5 describes the simulation test bed and Section 7.7 presents the methodology of the empirical study. Section 7.8 presents the experimental procedure of the experiments and Section 7.9 presents and discusses the result of the experiments. Section 7.10 summarizes the chapter.

7.2 Motivating Example

The following scenario serves as the motivating example for the empirical case study:

> *A customer (i.e. service requester) receives an SLA offer after discovering a suitable service from the service directory. Assuming that the customer has no or very little prior information about the service or the service provider, the customer assigns an initial belief about the probability of compliance of the SLA offer. The customer needs to find a way based on his/her belief plus some other mechanism in order to make decision whether to accept or reject the offer. The customer also need to assign utility value for each of the possible action, which will be used in the uncertainty tolerance mechanism.*

7.3 Study Requirements

The empirical study should demonstrate the following requirements:

1. **Uncertainty tolerance**: the study should demonstrate the ability of the uncertainty tolerance mechanism to tolerate the uncertainty in service provisioning offer (i.e. the service requester's initial degree of belief).

2. **Negative Consequence**: the study should demonstrate the potential negative consequence of the uncertainty in service provisioning offer.

3. **Positive Outcome**: the study should demonstrate that the uncertainty tolerance mechanism can overcome the uncertainty problem resulting in some measurable positive outcome to the service requester.

4. **Decision Making**: the study should demonstrate that the uncertainty tolerance mechanism must be able to assist the service requester in decision making, whether to accept or reject the SLA offer.

5. **Framework compliance**: the study should follow and implement the generic framework for uncertainty tolerance presented in Chapter 6.

7.4 Study Assumptions

There are several assumptions for the empirical study conducted in this chapter. These assumptions include assumptions made in previous chapters:

From Chapter 6:

- **rational agent:** the service requester entity is considered as a rational agent and is risk neutral. Under *Bayesianism*, a rational agent chooses the option with the highest expected utility.
- **trust assumption:** Trusted Service Provider and Evidence source: The framework operates under two trust assumptions: First, the SLA compliance status reported by the service provider after each service consumption is trusted, i.e. the service provider does not lie about the status of the SLA compliance. Secondly, the evidence collected from multiple sources is also trusted, i.e. the sources do not lie or untruthfully submit the evidence.

7.5 Testbed

The term ***testbed*** used in this study refers to a software-based platform which is used to run a simulation or experiment of a theory or research hypothesis. The testbed allows for running and replicating experiments within a close environment, protected from external factors such as network congestion and security issues. The term ***testbed*** has been used in many scientific areas and computer science as a reference to an environment to run experiments in a simulated manner whereby the actual execution of the experiments in real world can be costly, dangerous, or physically (hardware, etc.) not possible. Although the data and results collected from the testbed does not represent the real world, the information gathered through the analysis of the data will be useful in providing foundation for future works, and to validate the research question.

Since there us no existing simulation platform in the market that can provide the required functionalities (mainly generating the uncertainty, simulating customers request, and the uncertainty tolerance mechanism), the testbed used in this study has been design and developed from scratch. The testbed is developed using Java programming language (JDK 1.6) according to the uncertainty tolerance framework presented in Chapter 6. The main purpose of the testbed is to enable the simulation of a service provisioning environment (which involves service invocation) and to simulate the uncertainty in service provisioning offer. Subsequently, the testbed is also used to validate the uncertainty tolerance mechanism based on different parameters.

The testbed consists of several Java classes as follows:

- **Customer class:** this class represents the service requester entity.
- **Service class:** this class represents the service provider entity.
- **TrustedThirdParty class:** this class represents the trusted third party entity.
- **GatherEvidence class:** this class functions to collect evidence for the uncertainty tolerance mechanism in TrustedThirdParty class.
- **Simulation class:** this class acts as the driver class which run the entire simulation.

Apart from the above main classes, there are various other supporting classes for example for file input/output for data retrieval and data storage, graphing class, report class, and so on.

7.6 Experiment Parameters

There are several parameters of interest to the experiment:

- **service requester's initial degree of belief:** the value of this parameter is assigned based on the bootstrapping process as described in Chapter 4 and Chapter 5. The experiment will be conducted with different initial belief value to see the effect of the uncertainty tolerance mechanism on those values. Furthermore, different service requesters can assign different initial belief value for the same offer due to the subjective nature of the belief. Therefore, an experiment with two service requesters and different belief value will be conducted.
- **utility value:** the utility value assigned by a service requester will affect the outcome of the decision making process and also the cumulative utility parameter. Therefore, different experiments will be conducted to investigate the effect of different set utility value for a single service requester.

- **service failure level:** to simulate unstable service condition, different level of service failure is injected throughout the experiments.
- **number of *Accept* decision:** the number of *Accept* and *Reject* decisions can be compared between different sets of experiment execution with uncertainty and with uncertainty tolerance mechanism applied. In theory, the experiment with uncertainty tolerance should perform better, i.e. higher correct *Accept* decision.

7.7 Experiment Methodology

As part of the empirical study, a number of experiments are conducted to assess the viability of the uncertainty tolerance approach discussed in Chapter 5 and the uncertainty tolerance framework discussed in Chapter 6. This section describes the methodology which forms the basis of the empirical study.

The methodology for the empirical study is as follows:

- the initial state of the simulation is when the service requester (i.e. customer) has already found the required service from the service directory after going through the *search* and *discovery* process.
- the *utility values* and the *initial degree of belief* (i.e. prior) for the service requester are stored in a separate text file in CSV format.
- the service requester then initiates the uncertainty tolerance procedure which includes the evidence gathering, Bayes' inference, and decision making activities as described in Chapter 6. The decision making function will return a boolean value indicating whether the service requester should accept or reject the decision. This boolean value is stored for evaluation.
- to investigate the negative and positive consequence of the uncertainty tolerance mechanism, a base case is set with a condition that the service requester will always accept the offer, regardless of the outcome of the decision making process in previous step. The base case can then be compared to the situation whereby the service requester will evaluate the choice of action based on the expected utility value of each of the actions.

- to simulate service failure, random processing delays are injected during the experiment. The term *random* is used to reflect that the level of service failure is not constant over a period of time. The random numbers used to simulate the delays are generated using the Java random number generator class, *java.util.Random*. However, the Java random number generators are not truly random, they are algorithms that generate a fixed but random-looking sequence of numbers. There are several level of service failure based on how many failure is injected within a number of service invocations. The levels available are 10%, 20%, 40%, and 80%. For example, 10% service failure level indicates that during a period of N number of service invocations, 10% of N number of service invocations will fail.
- in order to show the negative consequence of uncertainty and the positive outcome of the uncertainty tolerance mechanism, a utility value termed as *cumulative utility* is introduced for the empirical study. The *cumulative utility* is the sum of utility after a number of independent service consumptions.
- once the service has completed and validated, the service requester receives a report for that particular service invocation from the service provider.

7.8 Experimental Procedure

Several sets of experiment are conducted to investigate the requirements described in Section 7.3. These experiments follows the methodology described in Section 7.7.

7.8.1 Experiment 1: Pessimistic Customer with 0.1 Degree of Belief

Based on earlier discussion in Chapter 4, *pessimistic customer* is a customer who has gone through the belief bootstrapping process and assigned a low belief value towards the service offer from the service provider. For this particular experiment, the following parameters are used:

- **pessimistic belief**: the prior is set to the value of 0.1 for each service offer.
- **utility value**: the utility values are set as shown in Table 7.1.

Table 7.1: Utility Values for Experiment 1

	Comply	Failure
Accept	10	-20
Reject	-5	-5

- **number of service invocations**: in order to show the consequence over a period of time, the number of service offer and subsequent service consumption is set to 1000 services. Each services is independent of each other.
- **service failure level**: the service level failure level is randomly varied over the whole experiment, over a set of service invocations.
- **evidence gathering parameters:** the number of other customers where the evidence is based upon and gathered is set to 4 customers and each customer is set to invoke the same number of services which is set to 10 services. These numbers (number of customers and number of services) are choosen as a representative of a possible scenario. These numbers are fixed throughout the experiment in order to simplify the experiment parameters and the execution of the simulation.
- **action taken:** for each service offer, the action taken by the customer is based upon the expected utility values of the set of possible actions (to accept or reject) as discussed in Chapter 5. This procedure applies to both untolerated and tolerated uncertainty scenario.
- **service metric:** for the purpose of this study, a single metric, *"execution time"* (as discussed in Section 6.5.3) is used as the quality indicator whihc is included in the SLA offer from the service provider. A value of 20ms is set in the simulation testbed for the process in a service execution (assuming each service execution only require a single process). This value is changed to a higher value during a service failure in order to simulate execution delay.

Various values such as the prior, posterior, expected utilities value, cumulative utility value are stored in text file using CSV format to be used in generating graph, as experiment's record, and for debugging purpose.

7.8.2 Experiment 2: Pessimistic Customer with 0.3 Degree of Belief

The objective of Experiment 2 is to investigate the different value of the prior which is set to 0.3 but still under the pessimistic view category. Other parameters are the same as in Experiment 1.

7.8.3 Experiment 3: Different Utility Sets for the Same Customer

The objective of Experiment 3 is to investigate the effect of different set of utility value for the same customer. In previous chapter, we have discussed and acknowledged that different set of utility value for the different action that can be taken by the customer can affect the outcome of the decision making process and in theory should affect the cumulative utility parameter.

The experiment is conducted with two different set of utility values for the same customer (as shown in Table 7.2 and Table 7.3), all other parameters are the same as in Experiment 1.

Table 7.2: Utility Set 1 for Exp. 3

	Comply	Failure
Accept	10	-20
Reject	-5	-5

Table 7.3: Utility Set 2 for Exp. 3

	Comply	Failure
Accept	5	-40
Reject	-10	-10

7.8.4 Experiment 4: Positive Outcome Based on Decision Making Process

The objective of Experiment 4 is to investigate the positive effect of the uncertainty tolerance mechanism through the decision taken by the service requester. The decision making process is determined by evaluating the expected utility value of the possible actions that can be taken by the service requester, i.e. whether to *Accept* or *Reject* the service provisioning offer. Since the calculation of the expected utility value is directly linked to the service requester's belief (in the form of probability value), therefore it is possible to investigate the positive outcome of the uncertainty tolerance mechanism by comparing one of the decision state of the service requester between the situation with uncertainty and with uncertainty tolerance mechanism applied.

The experiment is conducted with two different sets of utility value for the same customer (as shown in Table 7.4 and Table 7.5). Furthermore, different experiment will be conducted for each of the prior (uncertainty) values starting from 0.1 to 0.9 with a 0.2 increment. All other parameters are the same as in Experiment 1.

Table 7.4: Utility Set 1 for Exp. 4

	Comply	Failure
Accept	10	-20
Reject	-5	-5

Table 7.5: Utility Set 2 for Exp. 4

	Comply	Failure
Accept	5	-20
Reject	-20	-20

7.9 Empirical Results and Discussion

This section presents the experimental results and discussion for experiments conducted in Section 7.8.

7.9.1 Experiment 1 Result: Pessimistic Customer with 0.1 Degree of Belief

Figure 7.1 shows the result of Experiment 1. The top graph shows the plot of cumulative utility against number of service invocations. There are three trend lines on the top graph. The top trend line is the control case where it is the maximum cumulative utility if all services offer are accepted and comply (i.e. no service failure). The middle trend line shows the cumulative utility over a number of service invocations for the case of tolerated uncertainty (i.e. using posterior value), and the bottom trend line shows the cumulative utility over a number of service invocations for the case of untolerated uncertainty. The right hand side y-axis of the top graph represents the service failure level over a set of service invocations throughout the experiment. The value on the axis indicates the percentage of failure, for example, value of 4 represents 40% service failure level. The service failure level throughout the experiment is indicated using a box-like trend line on the graph.

The bottom graphs shows the three main values involve in the uncertainty tolerance mechanism, which are: (1) prior (uncertainty, i.e. customer's initial degree of belief), (2) evidence (gathered during the course of the experiment), and (3) posterior (tolerated

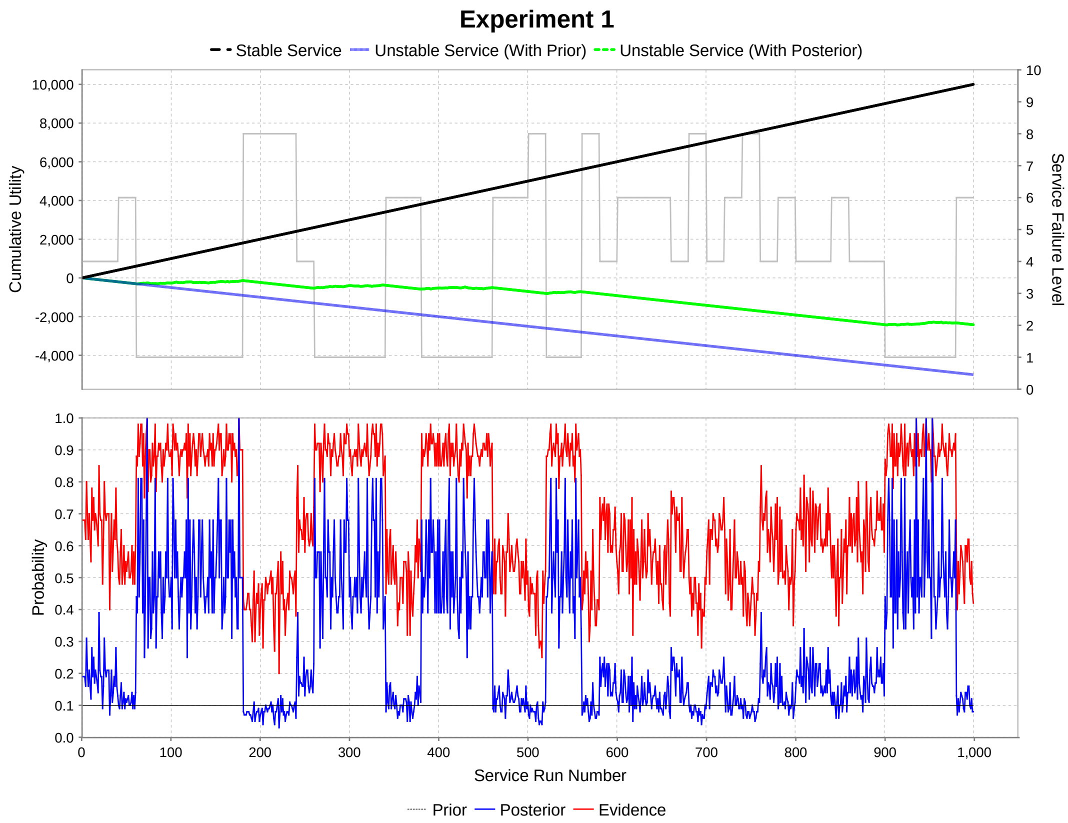
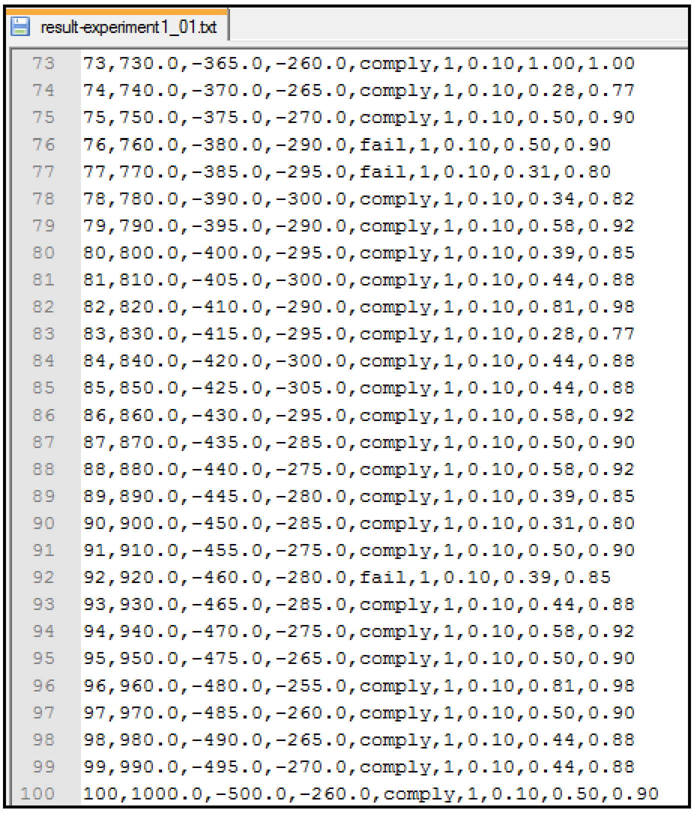


Figure 7.1: Pessimistic Customer with 0.1 Degree of Belief

uncertainty, i.e. updated customer's degree of belief). A snapshot of the experiment's parameters and results is shown in Figure 7.2.

Discussion: Experiment 1 result shows the requirements mentioned in Section 7.3 have been fulfilled. Firstly, for each service offer, the uncertainty tolerance mechanism is able to update customer's initial belief based on evidence gathering mechanism. This is shown in the snapshot of the statistic shown in Figure 7.2 and also the lower graph in Figure 7.1. For example, for service invocation number 86, the initial belief is 0.10 and is updated to 0.58 (posterior value, i.e. updated belief) based on evidence strength of 0.92. The bottom graph of Figure 7.1 shows that the service requester's initial belief is tolerated for each of the service invocation. In order to clearly view the different belief values, a snapshot of the experiment is shown in Figure 7.3, which shows the values of prior, posterior, and evidence from run number 400 to 450 (50 service invocation).


result-experiment1_01.txt

```
73,730.0,-365.0,-260.0,comply,1,0.10,1.00,1.00
74,740.0,-370.0,-265.0,comply,1,0.10,0.28,0.77
75,750.0,-375.0,-270.0,comply,1,0.10,0.50,0.90
76,760.0,-380.0,-290.0,fail,1,0.10,0.50,0.90
77,770.0,-385.0,-295.0,fail,1,0.10,0.31,0.80
78,780.0,-390.0,-300.0,comply,1,0.10,0.34,0.82
79,790.0,-395.0,-290.0,comply,1,0.10,0.58,0.92
80,800.0,-400.0,-295.0,comply,1,0.10,0.39,0.85
81,810.0,-405.0,-300.0,comply,1,0.10,0.44,0.88
82,820.0,-410.0,-290.0,comply,1,0.10,0.81,0.98
83,830.0,-415.0,-295.0,comply,1,0.10,0.28,0.77
84,840.0,-420.0,-300.0,comply,1,0.10,0.44,0.88
85,850.0,-425.0,-305.0,comply,1,0.10,0.44,0.88
86,860.0,-430.0,-295.0,comply,1,0.10,0.58,0.92
87,870.0,-435.0,-285.0,comply,1,0.10,0.50,0.90
88,880.0,-440.0,-275.0,comply,1,0.10,0.58,0.92
89,890.0,-445.0,-280.0,comply,1,0.10,0.39,0.85
90,900.0,-450.0,-285.0,comply,1,0.10,0.31,0.80
91,910.0,-455.0,-275.0,comply,1,0.10,0.50,0.90
92,920.0,-460.0,-280.0,fail,1,0.10,0.39,0.85
93,930.0,-465.0,-285.0,comply,1,0.10,0.44,0.88
94,940.0,-470.0,-275.0,comply,1,0.10,0.58,0.92
95,950.0,-475.0,-265.0,comply,1,0.10,0.50,0.90
96,960.0,-480.0,-255.0,comply,1,0.10,0.81,0.98
97,970.0,-485.0,-260.0,comply,1,0.10,0.50,0.90
98,980.0,-490.0,-265.0,comply,1,0.10,0.44,0.88
99,990.0,-495.0,-270.0,comply,1,0.10,0.44,0.88
100,1000.0,-500.0,-260.0,comply,1,0.10,0.50,0.90
```

Figure 7.2: Snapshot of Experiment 1 recorded parameters and results

The snapshot shows that service requester assigns the same initial belief value (since each service offer is independent of each other), and based on the evidence collected, the uncertainty tolerance mechanism updates the initial belief value to produce the posterior value (i.e. tolerated uncertainty).

Secondly, Experiment 1 result also demonstrates both the negative consequence of the uncertainty and positive outcome of the uncertainty tolerance mechanism. For example, for the negative consequence, the cumulative utility value for the case of untolerated uncertainty after 1000 service invocation is -5000 while the cumulative utility value for the tolerated uncertainty is -2420. The cumulative utility value for the tolerated uncertainty is better (lower negative value, i.e. lower loss) compared to the untolerated case.

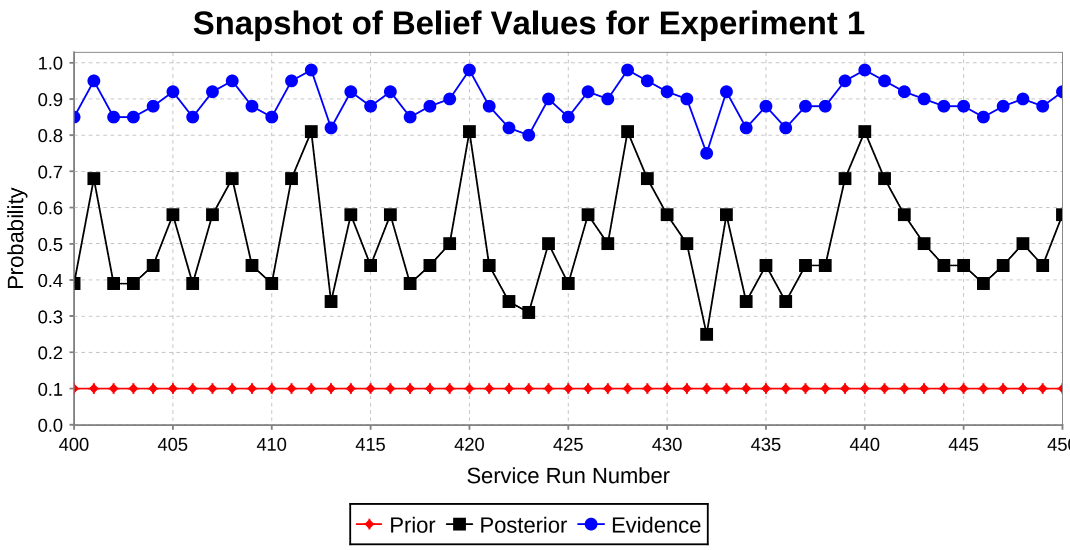


Figure 7.3: Snapshot of Experiment 1 Belief Values

7.9.2 Experiment 2 Result: Pessimistic Customer with 0.3 Degree of Belief

Figure 7.4 shows the graphs for the result of Experiment 2. The format of graphs is as discussed in Experiment 1 result.

Discussion: The result for Experiment 2 is consistent with the requirements discussed in Section 7.3 and also consistent with the result of Experiment 1. The cumulative utility value of the tolerated uncertainty is 745 which is comparatively better than the cumulative utility value for the untolerated uncertainty which is -5000.

The conclusion for Experiment 1 and 2 is that the uncertainty tolerance mechanism is able to tolerate customer's belief uncertainty, when the customer inadvertently assign inaccurate initial degree of belief.

7.9.3 Experiment 3 Result: Different Utility Sets for the Same Customer

Figure 7.5a shows the graph for the result for a customer with utility set 1 and Figure 7.5b shows the graph for customer with utility set 2. The format of graphs is as discussed in Experiment 1 result.

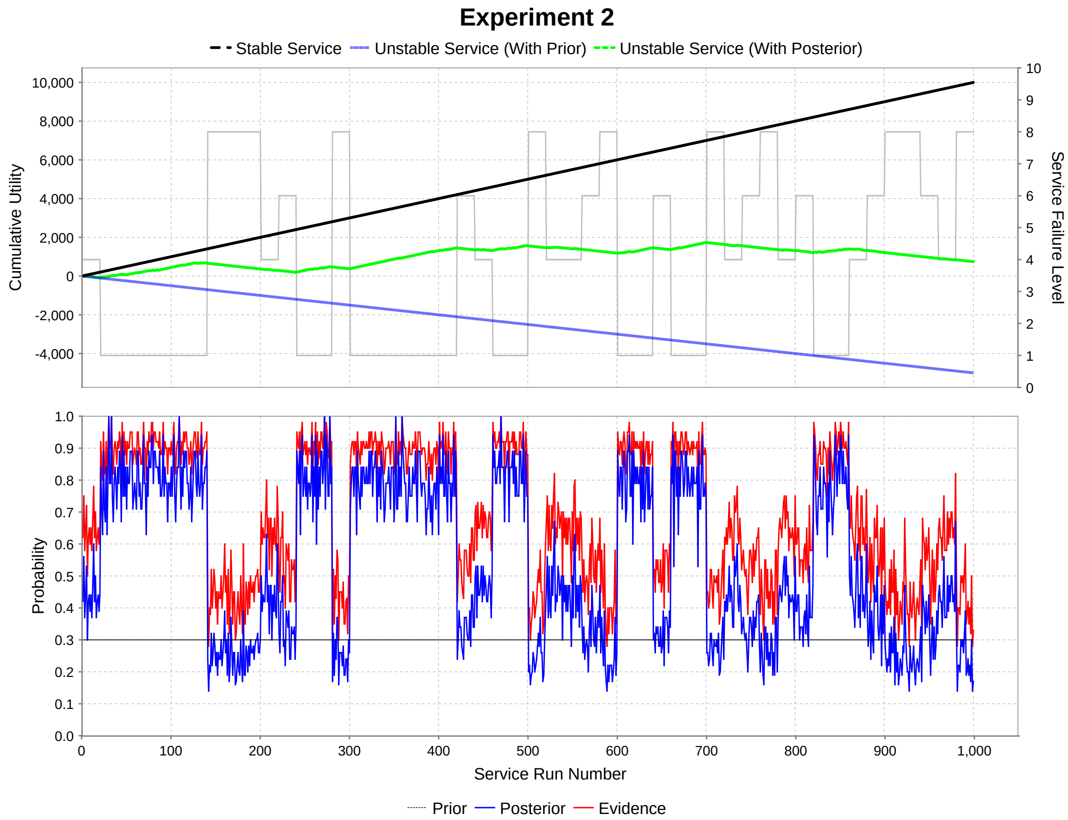
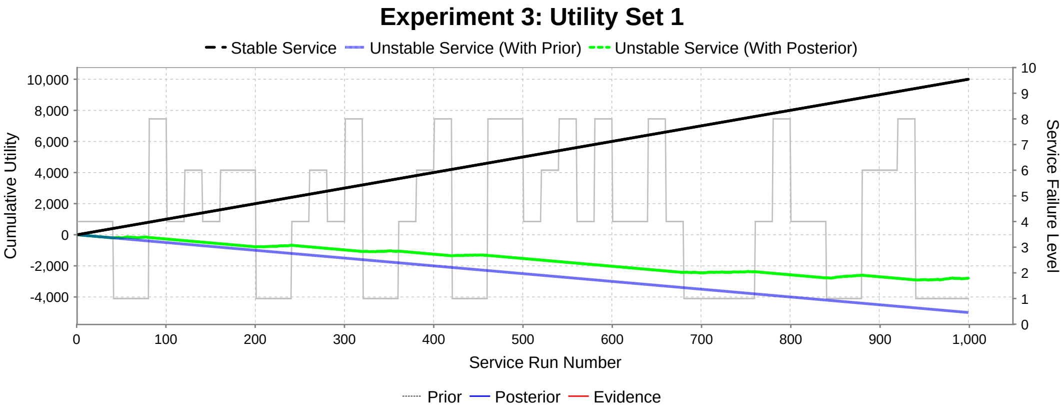
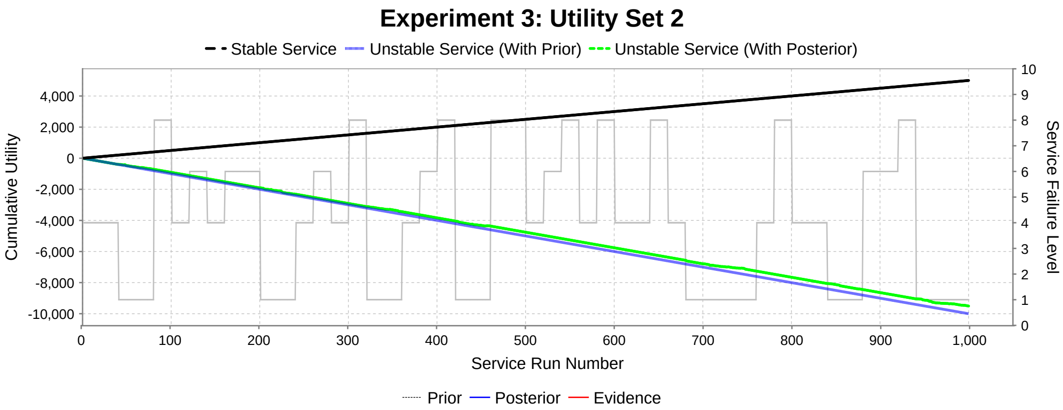


Figure 7.4: Pessimistic Customer with 0.3 Degree of Belief

Discussion: The graph for utility set 1 (Figure 7.5a) shows that the final value of total cumulative utility for the tolerated uncertainty is -2795 and the untolerated case is -5000. On the other hand, the graph for utility set 2 (Figure 7.5b) shows that the final value for the final value of the total cumulative utility for the tolerated uncertainty is -9505 and the untolerated case is -10000. The results clearly show that different set of utility value for the same customer under the same operating parameters (service level fault, initial belief value, and evidence gathering parameters) affect the outcome of the uncertainty tolerance mechanism in the form of the total cumulative value. The same conclusion can also be inferred for the case of different customers (i.e. two customer) with different utility set under the same operating environment.



(a) Customer A with Utility Set 1



(b) Customer A with Utility Set 2

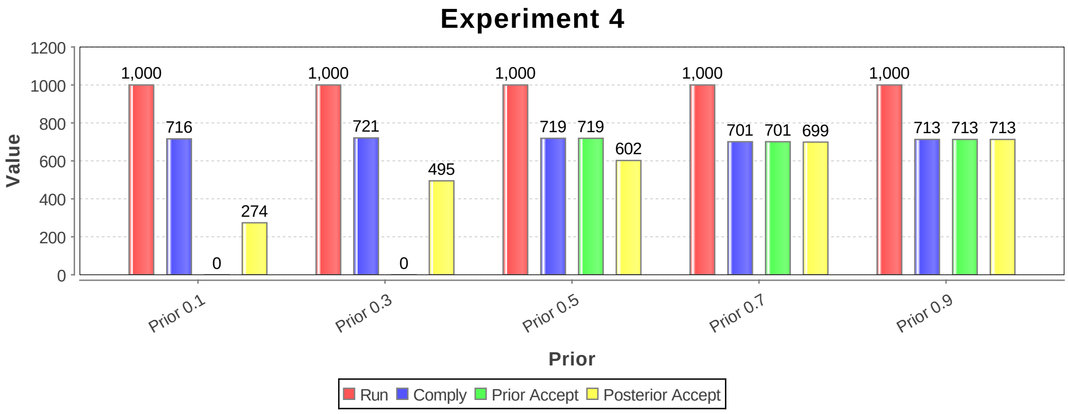
Figure 7.5: Experiment 4: Positive Outcome Based on Decision Making Process

7.9.4 Experiment 4: Positive Outcome Based on Decision Making Process

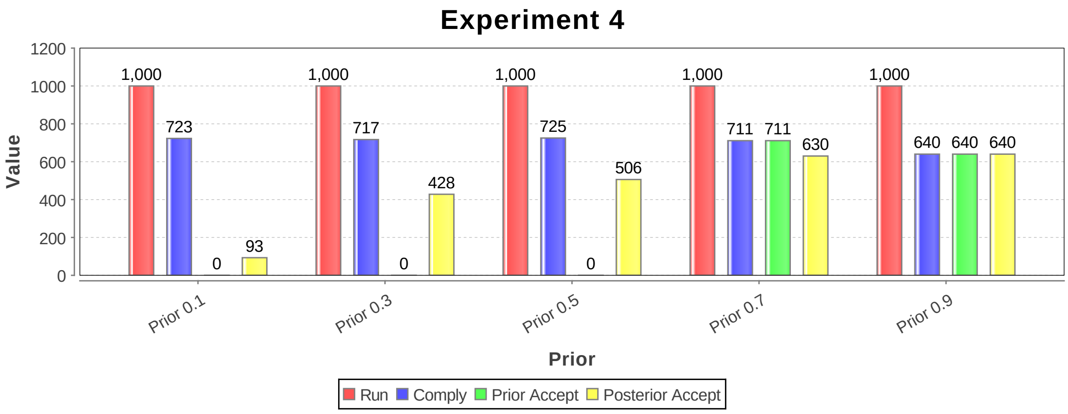
Figure 7.6a shows the graph for the result for a customer with utility set 1 and Figure 7.6b shows the graph for customer with utility set 2. The format of the graphs is as follows: the y-axis represents the number of service execution in the experiment and the x-axis represents the category of the uncertainty using prior value as the quantitative representative. There are 5 categories, each is mapped to specific prior value, starting from 0.1 up to 0.9 with 0.2 increment. These categories represent the possible belief assignment for the service requester through the bootstrapping mechanism as discussed in Section 4.5.4. For each category, there are 4 indicators: (1) the first indicator is the total number of service executions for the experiment which is set to 1000 for all categories, this serves as the base number for comparison for the other indicators, (2) the second indicator is the *"comply"* status of each service execution. The next two are the key indicators for this experiment, (3) the third indicator is the *"Accept"* rate for the service offer with uncertainty (prior), and the (4) fourth indicator is the *"Accept"* rate for the service offer with uncertainty tolerance mechanism applied (posterior).

Discussion: In theory, the service offers with uncertainty tolerance mechanism applied should produce better acceptance rate as compared to the service offers with uncertainty. For example, referring to Figure 7.6a, category *Prior 0.1*, the total number of service execution is 1000, and out of that 716 service executions comply with the metric stated in the SLA (metric similar to Experiment 1). As for the case of service offer with uncertainty, using the prior value in the expected utility calculation resulted in none of the service is *"Accept"*-ed by the service requester. On the other hand, the service offer with the uncertainty tolerance mechanism applied yielded 271 *"Accept"* which corresponds to 38.3% of the total number of service executions that comply with the service SLA compliance.

This positive outcome is also reflected for category *Prior 0.3*, whereby the number of service compliance is 721, number of *"Accept"* for uncertainty case is none, and number of *"Accept"* for uncertainty tolerance case is 495. Due to the nature of the expected utility calculation which is linked to the utility value, the value of number of *"Accept"* for uncertainty case in category *Prior 0.5* is slightly lower than the uncertainty case.



(a) Customer A with Utility Set 1



(b) Customer A with Utility Set 2

Figure 7.6: Experiment 4: Positive Outcome Based on Decision Making Process

As for category *Prior 0.7* and category *Prior 0.9*, the difference is minimal or the same between the two cases.

Furthermore, Figure 7.6b shows the graph for customer with utility set 2. We can see that a similar trend of positive outcome as in experiment with utility set 1, whereby in category *Prior 0.1*, *Prior 0.3*, and *Prior 0.5*, the number of *"Accept"* cases for uncertainty tolerance is higher than the service offer with uncertainty. As for category *Prior 0.7* and category *Prior 0.9*, the difference is minimal or the same between the two cases.

7.10 Summary

This chapter presents an empirical study of uncertainty in service provisioning offer that affect a service requester. The empirical study is carried out based on a simulation testbed that has been implemented following the uncertainty tolerance framework presented in Chapter 6 and based on the approach to uncertainty tolerance presented in Chapter 5. The contribution of this chapter is the validation of the viability of the uncertainty mechanism and its subsequent positive outcome to the service requester. The result from various experiments does indicate that the the uncertainty tolerance mechanism is a viable solution to solve the uncertainty issue in service provisioning offer, and in general validate the uncertainty tolerance approach presented in Chapter 5. In the next chapter, we will summarize the research, the contributions of the research and potential future work.

Chapter 8

Conclusion and Future Work

This thesis has presented the overall issue of uncertainty in service provisioning environment from the service requester perspective and proposed the concept of uncertainty tolerance. The main focus of this research is to address the issue of uncertainty faced by a service requester (consumer or end user) when presented with an offer of Service Level Agreement (SLA) from a service provider. An approach to tolerate uncertainty in service provisioning offer plus a generic framework to implement the approach have also been proposed. The proposed uncertainty tolerance approach utilizes the subjective probability framework in the form of Bayes' Theorem, evidence gathering mechanism and followed by decision making through expected utility value. The empirical study conducted has shown the viability of the proposed approach and framework, and also have shown the positive outcome of such approach.

Although the focus of this research is to address the issue of uncertainty from the service requester point of view, the idea of addressing uncertainty in service provisioning lifecycle can also benefits the service providers as well. For example, when a service requester discovered a service registered in a service directory, the service requester will request the service provider to provide or offers the Service Level Agreement to the service requester. At this point service providers can utilize the uncertainty tolerance framework as follows.

Using the uncertainty tolerance framework, the service provider can determine a suitable SLA based on the current performance based on the value provided by the framework. In essence, the framework can be used by the service provider as a self assessment tool to provide suitable SLA, and also can be a tool to negotiate the proposed SLA with the service requester.

The remainder of this chapter is organized as follows: Section 8.1 highlights the contributions made by this thesis. Section 8.2 provides discussion on potential future works which can be derived from the research works conducted in this thesis.

8.1 Summary of Contributions

The thesis has made several contributions as follows:

- **Classification scheme of uncertainty in service provisioning (Chapter 4**: Based on the insight to the issue of uncertainty in other areas such as health care and management, a classification scheme consists of different dimensions such as *temporal*, *entity*, *nature*, and *level* has been proposed for uncertainty in the area of service provisioning. Furthermore, a multidimensional classification scheme based on the four dimensions is introduced, since for most uncertainty in service provisioning, a single dimension does not provides accurate classification of the uncertainty. To our knowledge, the proposed classification scheme is unique and has not been introduced by other researchers. The value of such classification scheme is to provide interested parties such as researchers, system designer, and system developer with a tool to accurately classify uncertainty in a service provisioning system.

- **A view of uncertainty in service provisioning**: One of the major challenges concerning the issue of uncertainty in service provisioning is the inconsistency in representing the uncertainty from a single unified perspective. This is due to the subjective nature of the problem (uncertainty) that leads to possible different representation from the perspective of different interested parties. Therefore, a view on uncertainty based on temporal classification scheme, linked with the service requester's perspective has been proposed in Chapter 4. The view provides a logical and easy to understand representation of uncertainty in service provisioning. This approach differs from previous work since it provides a complete view of uncertainty in a service provisioning process whereby previous works only addresses specific uncertainty problem. Additionally, in contrast to previous work, the proposed view is tightly coupled with service requester's perspective as compared to the previous work which concentrates more on service provider's view of uncertainty. The value of the proposed view is a clear understanding of the problem of uncertainty in service provisioning which provides a groundwork for selecting appropriate measure to overcome the uncertainty issue.

- **A generic concept of uncertainty tolerance in service provisioning**: Based on the temporal view representation of uncertainty, linked with the service requester's perspective, a generic concept for uncertainty tolerance has been proposed (Chapter 4. This concept relies on the fact that the uncertainty issue across a service provisioning lifecycle is due to the possible inaccurate initial belief of the service requester. This is caused by lack of information or gap in knowledge of the service requester. Therefore, the uncertainty tolerance concept relies on approaches that is able to reduce this gap of knowledge by gathering information from external sources.

- **An approach to uncertainty tolerance in service provisioning offer**: Selecting the problem of uncertainty in service provisioning offer (in the pre service provisioning phase) as a proof of concept, an approach to tolerate uncertainty based on the combination of subjective probability framework, evidence gathering and expected utility values has been proposed. This approach, in contrast to previous work, allows a service requester to initially express or assign an initial degree of belief to the service offer from a service provider. The value of this approach is to minimize the negative consequence of the uncertainty, and enable decision making process of the service requester. Furthermore, such approach will provide confidence to the service requester towards the service offer from the service provider.

- **A generic framework for uncertainty tolerance in service provisioning offer**: The proposed framework represents a generic application level blueprint for implementing the uncertainty tolerance concept. This approach is unique and important since there has been no such attempt to provide a framework that addresses the issue of uncertainty in service provisioning offer. The framework can be a valuable tool to interested parties such as system designer and developer of service-based system. The framework will enable a service-based system to be designed to be able to tolerate the problem of uncertainty faced by a service requester.

8.2 Future Work

The research work presented in this thesis provides a basis for a number of potential related future works as follows:

- **End-to-end coverage of uncertainty tolerance**: The approach towards uncertainty tolerance presented in Chapter 5, the generic framework presented in Chapter 6 and the empirical study conducted in Chapter 7 addresses the issue of uncertainty in the pre-sp phase which concerns with the service offer from the service provider. Future work should consider an end-to-end coverage of the uncertainty tolerance mechanism, i.e. from the start to the end of the service provisioning lifecycle. The challenge is that different types of uncertainty in different phases of the lifecycle might require different approach of uncertainty tolerance. The main challenge for implementing start-to-end uncertainty tolerance for a service provisioning system is that it could lead to significant level of complexity to the solution.

- **Cost of uncertainty tolerance**: The proposed uncertainty tolerance mechanism has associated cost (time, financial, resources, etc.) which has been discussed but not quantitatively investigated. Future work should include cost consideration in the implementation of the uncertainty tolerance mechanism so that the benefit of the uncertainty tolerance mechanism can be balanced with the associated cost of the measure taken.

- **Multiple metrics**: The empirical study conducted in Chapter 7 only considers a single metric for SLA violation. This does not reflect real world requirement of a service provisioning system whereby more than one metric is in consideration by the service requester. The challenge is that additional metrics will increase the complexity of the uncertainty problem significantly, resulting in added complexity to the uncertainty tolerance mechanism.

- **Prototype implementation**: A potential thread for future work is to implement the proposed uncertainty tolerance mechanism and the generic framework in a real world service provisioning system such as web services based service provisioning system. Such implementation will provide an insight on the effectiveness of the solution in term of the viability (cost related) and performance.

Bibliography

[1] Tom Bellwood, Luc Clément, David Ehnebuske, Andrew Hately, Maryann Hondo, Yin Leng Husband, Karsten Januszewski, Sam Lee, Barbara McKee, Joel Munter, et al. **UDDI Version 3.0**. *Published specification, Oasis*, **5**:16–18, 2002. ix, 27

[2] John M. Rathmell. **What Is Meant by Services?** *The Journal of Marketing*, **30**(4):32–36, 1966. xi, 18, 19, 21, 22, 24

[3] Roland T. Rust and Carol Miu. **What academic research tells us about service**. *Communications of The ACM*, **49**:49–54, 2006. 1

[4] Linda Dailey Paulson. **Services Science: A New Field for Today's Economy**. *Computer*, **39**:18–21, 2006. 1

[5] L. J. Zhang. **EIC Editorial: Introduction to the Knowledge Areas of Services Computing**. *IEEE Trans. Services Computing*, **1**(2):62–74, 2008. 2, 18

[6] M.N. Huhns and M.P. Singh. **Service-oriented Computing: Key Concepts and Principles**. *IEEE Internet Computing*, **9**(1):75–81, January 2005. 2

[7] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. **Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility**. *Future Generation Computer Systems*, **25**(6):599–616, June 2009. 2

[8] K Mockford. **Web Services Architecture**. *BT Technology Journal*, **22**(1):19–26, 2004. 2

[9] M. A. Rappa. **The utility business model and the future of computing services**. *IBM Systems Journal*, **43**(1):32–42, 2004. 2

[10] Andrew S. Tanenbaum and Maarten van Steen. *Distributed Systems: Principles and Paradigms*. Prentice Hall, September 2001. 2

[11] Mike P. Papazoglou. **Service-Oriented Computing: Concepts, Characteristics and Directions**. In *Web Information Systems Engineering*, pages 3–12, 2003. 2

[12] Kemal A. Delic and Martin Anthony Walker. **Emergence of the Academic Computing Clouds**. *Ubiquity*, 2008. 2

[13] I. Foster, Yong Zhao, I. Raicu, and S. Lu. **Cloud Computing and Grid Computing 360-Degree Compared**. In *Grid Computing Environments Workshop, 2008. GCE '08*, pages 1 –10, nov. 2008. 2

[14] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, and J. Brobergand I. Brandic. **Cloud Computing and Emerging IT Platforms: Vision, hype, and reality for delivering computing as the 5th utility**. *Future Generation Computer Systems*, **25**:599–616, 2009. 2

[15] Ruth Mortimer. **A silver lining in the clouds**, January 2012. 2

[16] quality of service. **BusinessDictionary.com**, 2013. Online; Accessed 10-April-2013, Available from http://www.businessdictionary.com/definition/quality-of-service-QoS.html. 3

[17] **ITU-T Recommendation E.800. Quality of Telecommunication Services: Concepts, Models, Objectives and Dependability Planning. Terms and Definitions Related to the Quality of Telecommunication Services**, September 2008. 3

[18] H. Ludwig, A. Keller, A. Dan, and R. King. **A service level agreement language for dynamic electronic services**. In *Advanced Issues of E-Commerce and Web-Based Information Systems, 2002. (WECWIS 2002). Proceedings. Fourth IEEE International Workshop on*, pages 25 – 32, 2002. 3, 33

[19] D. C. Verma. **Service level agreements on IP networks**. *Proceedings of the IEEE*, **92**(9):1382–1388, 2004. 3

[20] E. Marilly, O. Martinot, H. Papini, and D. Goderis. **Service level agreements: a main challenge for next generation networks**. In *Universal Multiservice Networks, 2002. ECUMN 2002. 2nd European Conference on*, pages 297 – 304, 2002. 3

[21] Zhang Nan, X. Qiu, and L. Meng. *A SLA-Based Service Process Management Approach for SOA*. IEEE, October 2006. 3

[22] M.P. Papazoglou and W.-J. van den Heuvel. **Web Services Management: A Survey**. *IEEE Internet Computing*, **9**(6):58–64, November 2005. 3

[23] Johari Abdullah and Aad van Moorsel. **Uncertainty and Uncertainty Tolerance in Service Provisioning**. *Journal of Internet Services and Information Security (JISIS)*, **1**(4):89–109, 11 2011. 7

[24] Hemant K. Bhargava and Shankar Sundaresan. **Contingency Pricing for Information Goods and Services Under Industrywide Performance Standard**. *J. Manage. Inf. Syst.*, **20**(2):113–136, October 2003. 10

[25] Max H. Bazerman and James J. Gillespie. **Betting on the Future: The Virtues of Contingent Contracts.** *Harvard Business Review*, **77**(5):155 – 160, 1999. 10, 11

[26] Chris Smith and Aad van Moorsel. **Mitigating Provider Uncertainty in Service Provision Contracts**. In Dirk Neumann, Mark Baker, Jörn Altmann, and Omer Rana, editors, *Economic Models and Algorithms for Distributed Systems*, pages 143–159. Birkhäuser Basel, 2010. 10

[27] Eyal Biyalogorsky and Eitan Gerstner. **Contingent Pricing to Reduce Price Risks**. *Marketing Science*, **23**:146–155, 2004. 11

[28] Scott Fay and Jinhong Xie. **The Economics of Buyer Uncertainty: Advance Selling vs. Probabilistic Selling**. *Marketing Science*, **29**:1040–1057, 2010. 11

[29] H.K. Bhargava and S. Sundaresan. **Managing quality uncertainty through contingency pricing**. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, page 9 pp., jan. 2003. 11

[30] George A. Akerlof. **The Market for "Lemons": Quality Uncertainty and the Market Mechanism**. *The Quarterly Journal of Economics*, **84**(3):pp. 488–500, 1970. 11

[31] Curtis R. Reitz. *Consumer protection under the Magnuson-Moss Warranty Act*. American Law Institute-American Bar Association Committee on Continuing Professional Education (Philadelphia, Pa.), 1978. 12

[32] V Padmanabhan and Ram C Rao. **Warranty policy and extended service contracts: Theory and an application to automobiles**. *Marketing Science*, **12**(3):230–247, 1993. 12

[33] Shiou Shieh. **Price and Money-Back Guarantees as Signals of Product Quality**. *Journal of Economics & Management Strategy*, **5**(3):361–377, 1996. 12

[34] Scott Davis, Eitan Gerstner, and Michael Hagerty. **Money back guarantees in retailing: matching products to consumer tastes**. *Journal of Retailing*, **71**(1):7 – 22, 1995. 12

[35] Amir Heiman, Bruce McWilliams, and David Zilberman. **Demonstrations and money-back guarantees: market mechanisms to reduce uncertainty**. *Journal of Business Research*, **54**(1):71 – 84, 2001. 12

[36] Paul A. Pavlou, Huigang Liang, and Yajiong Xue. **Understanding and mitigating uncertainty in online exchange relationships: a principal- agent perspective**. *MIS Q.*, **31**:105–136, March 2007. 12

[37] Ray Rees. **The Theory of Principal and Agent: Part 1**. *Bulletin of Economic Research*, **37**(1):3–26, January 1985. 12

[38] Ray Rees. **The Theory of Principal and Agent: Part 2**. *Bulletin of Economic Research*, **37**(2):75–95, 1985. 12

[39] John H Huston and Roger W Spencer. **Quality , uncertainty and the internet : The market for cyber lemons**. *Spring*, 2002. 12

[40] Yong Pan. **Improving the Cyber 'Lemons' Market with a Trust-Intermediary in E-Commerce: Model and Case Study**. In *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, pages 1 –6, oct. 2008. 12

[41] Yong Pan. **Eliminating the cyber 'lemons' problem with the e-reputation in e-commerce market: theoretical model and practice**. *IJNVO*, **8**(3/4):182–191, 2011. 12

[42] Torsten Eymann, Stefan Knig, and Raimund Matros. **A Framework for Trust and Reputation in Grid Environments**. *Journal of Grid Computing*, **6**:225–237, 2008. 12

[43] Lea Viljanen. **Towards an Ontology of Trust**. In Sokratis Katsikas, Javier Lopez, and Günther Pernul, editors, *Trust, Privacy and Security in Digital Business*, **3592** of *Lecture Notes in Computer Science*, pages 175–184. Springer Berlin / Heidelberg, 2005. 13, 57

[44] Sini Ruohomaa and Lea Kutvonen. **Trust Management Survey**. In Peter Herrmann, Valérie Issarny, and Simon Shiu, editors, *Trust Management*, **3477** of *Lecture Notes in Computer Science*, pages 77–92. Springer Berlin / Heidelberg, 2005. 13

[45] Colin English, Sotirios Terzis, and Waleed Wagealla. *Engineering Trust Based Collaborations in a Global Computing Environment.* 2004. 13

[46] Sviatoslav Brainov and Tuomas Sandholm. **Contracting with uncertain level of trust**. In *Proceedings of the 1st ACM conference on Electronic commerce*, EC '99, pages 15–21, New York, NY, USA, 1999. ACM. 13

[47] Arthur P Dempster. **Upper and lower probabilities induced by a multivalued mapping**. *The annals of mathematical statistics*, **38**(2):325–339, 1967. 14

[48] Glenn Shafer. *A mathematical theory of evidence*, **1**. Princeton university press Princeton, 1976. 14

[49] Jeffrey A Barnett. **Computational methods for a mathematical theory of evidence**. In *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pages 197–216. Springer, 2008. 14

[50] Malcolm Beynon, Bruce Curry, and Peter Morgan. **The DempsterShafer theory of evidence: an alternative approach to multicriteria decision modelling**. *Omega-international Journal of Management Science*, **28**:37–50, 2000. 14, 15

[51] Xiujuan Gao, Hongxun Yao, Wen Gao, and Wei Zeng. *Fusion of Biometrics Based on DS Theory*. 2001. 14

[52] S. Foucher and L. Gagnon. **Face recognition in video using Dempster-Shafer theory**. In *International Conference on Acoustics, Speech, and Signal Processing*, **3**, 2004. 14

[53] H. Leung and Jiangfeng Wu. **Bayesian and Dempster-Shafer target identification for radar surveillance**. *IEEE Transactions on Aerospace and Electronic Systems*, **36**:432–447, 2000. 14

[54] Thomas M. Chen and Varadharajan Venkataramanan. **Dempster-Shafer Theory for Intrusion Detection in Ad Hoc Networks**. *IEEE Internet Computing*, **9**:35–41, 2005. 14

[55] Wei Hu, Jianhua Li, and Qiang Gao. **Intrusion Detection Engine Based on Dempster-Shafer's Theory of Evidence**. In *International Conference on Communications, Circuits and Systems*, 2006. 14

[56] Ewa Straszecka. **Combining uncertainty and imprecision in models of medical diagnosis**. *Information Sciences*, **176**:3026–3059, 2006. 14

[57] James C Hoffman and Robin R Murphy. **Comparison of Bayesian and Dempster-Shafer theory for sensing: a practitioner's approach**. In *SPIE's 1993 International Symposium on Optics, Imaging, and Instrumentation*, pages 266–279. International Society for Optics and Photonics, 1993. 14

[58] Thomas L Saaty. *What is the analytic hierarchy process?* Springer, 1988. 14

[59] Thomas L Saaty. **Analytic hierarchy process**. *Encyclopedia of Biostatistics*, 2005. 14

[60] Thomas L. Saaty. **How to make a decision: The Analytic Hierarchy Process**. *European Journal of Operational Research*, 1994. 15

[61] Robert Handfield, Steven V Walton, Robert Sroufe, and Steven A Melnyk. **Applying environmental criteria to supplier assessment: A study in the application of the Analytical Hierarchy Process**. *European Journal of Operational Research*, **141**(1):70–87, 2002. 15

[62] Fariborz Y Partovi, Jonathan Burton, and Avijit Banerjee. **Application of analytical hierarchy process in operations management**. *International Journal of Operations & Production Management*, **10**(3):5–19, 1990. 15

[63] Kamal M Al-Harbi. **Application of the AHP in project management**. *International journal of project management*, **19**(1):19–27, 2001. 15

[64] James S Dyer. **Remarks on the analytic hierarchy process**. *Management science*, **36**(3):249–258, 1990. 15

[65] Anjali Awasthi and Satyaveer S. Chauhan. **Using AHP and DempsterShafer theory for evaluating sustainable transport solutions**. *Environmental Modelling and Software*, **26**:787–796, 2011. 15

[66] Ye Qing, Zhang Changhong, Wu Xiaoping, and Zhai Dingjun. **Information Security Risk Assessment Based on AHP/DST**. In *International Conference on Management and Service Science*, 2009. 15

[67] Lu Simei, Zhang Jianlin, Sun Hao, and Luo Liming. **Security Risk Assessment Model Based on AHP/DS Evidence Theory**. In *International Forum on Information Technology and Applications*, 2009. 15

[68] Gang Wei and Yang Liu. **A DS/AHP method for comprehensive decision-making in urban power system planning**. In *Electricity Distribution, 2008. CICED 2008. China International Conference on*, pages 1–5, 2008. 15

[69] Susumu Konnoy, Abar Sameeray, Yukio Iwayaz, Toru Abey, and Tetsuo Kinoshitay. **Knowledge-Based Support of Network Management Tasks Using Active Information Resource**. In *International Agent Technology Conference*, pages 195–199, 2006. 15

[70] Qiang Guan, Ziming Zhang, and Song Fu. **Ensemble of Bayesian Predictors for Autonomic Failure Management in Cloud Computing**. In *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*, pages 1–6, July 2011. 15

[71] A. Ghaffarkhah and Y. Mostofi. **Channel learning and communication-aware motion planning in mobile networks**. In *American Control Conference (ACC), 2010*, pages 5413–5420, June 2010. 15

[72] P. Chemouil, J. Khalfet, and M. Lebourges. **A fuzzy control approach for adaptive traffic routing**. *Communications Magazine, IEEE*, **33**(7):70–76, Jul 1995. 16

[73] Yongyi Mao, F.R. Kschischang, Baochun Li, and S. Pasupathy. **A factor graph approach to link loss monitoring in wireless sensor networks**. *Selected Areas in Communications, IEEE Journal on*, **23**(4):820–829, April 2005. 16

[74] Ming Ni, J.D. McCalley, V. Vittal, and T. Tayyib. **Online risk-based security assessment**. *Power Systems, IEEE Transactions on*, **18**(1):258–265, Feb 2003. 16

[75] Services Computing Community. **IEEE Online Community: Services Computing Community**, 2012. Online; Accessed 19-April-2012. 18

[76] SCC 2004. **2004 IEEE International Conference on Services Computing**, 2004. Online; Accessed 19-April-2012. 18

[77] Christopher H. Lovelock. **Classifying Services to Gain Strategic Marketing Insights**. *The Journal of Marketing*, **47**(3):pp. 9–20, 1983. 19

[78] William J. Regan. **The Service Revolution**. *The Journal of Marketing*, **27**(3):pp. 57–62, 1963. 19, 21

[79] G. Lynn Shostack. **Breaking Free from Product Marketing**. *The Journal of Marketing*, **41**(2):pp. 73–80, 1977. 19, 21

[80] Kenneth K. Boyer, Roger Hallowell, and Aleda V. Roth. **E-services: operating strategy–a case study and a method for analyzing operational benefits**. *Journal of Operations Management*, **20**(2):175 – 188, 2002. 23, 24

[81] Jennifer Rowley. **An analysis of the e-service literature: towards a research agenda**. *Internet Research*, **16**(3):339–359, 2006. 23, 24

[82] Schahram Dustdar and Wolfgang Schreiner. **A survey on web services composition**. *Int. J. Web Grid Serv.*, **1**:1–30, August 2005. 23, 24

[83] David Booth, Francis McCabe, and Michael Champion. **Web Services Architecture - W3C Working Group Note 11 February 2004**. Technical report. 23, 24, 102, 105

[84] D. Kuebler and W. Eibach. **Metering and accounting for Web services**, July 2001. 23

[85] Liang-Jie Zhang, Jia Zhang, and Hong Cai. *Services Computing*. Springer, 2007. 23, 24

[86] AWS. **Amazon Web Services**, 2012. Online; Accessed 19-April-2012. 25

[87] SalesForce. **SalesForce**, 2012. Online; Accessed 19-April-2012. 25

[88] Harry Chen, Dipanjan Chakraborty, Liang Xu, Anupam Joshi, and Tim Finin. **Service Discovery in the Future Electronic Market**. 2000. 25

[89] Fabio Casati, Ski Ilnicki, Li jie Jin, Vasudev Krishnamoorthy, and Ming chien Shan. *Adaptive and Dynamic Service Composition in eFlow*. 2000. 25

[90] H. Chi Wong and Katia P. Sycara. **A Taxonomy of Middle-Agents for the Internet**. In *International Conference on Multiagent Systems*, pages 465–466, 2000. 25

[91] Keith Decker, Mike Williamson, and Katia Sycara. **Matchmaking and Brokering**. 1996. 25

[92] Robin M. E. Swezey, Masato Nakamura, Shun Shiramatsu, Tadachika Ozono, and Toramatsu Shintani. *Architecture for Automated Search and Negotiation in Affiliation among Community Websites and Blogs*. 2010. 25

[93] Özgür Gümüs, Önder Gürcan, and Oguz Dikenelli. *Towards a Broker Agent in the Semantic Services Environment*. 2008. 25

[94] Christian Grönroos. **A Service-Orientated Approach to Marketing of Services**. *European Journal of Marketing*, **12**:588–601, 1978. 28

[95] Christian Grönroos. **An Applied Service Marketing Theory**. *European Journal of Marketing*, **16**:30–41, 1982. 28

[96] R. C. Lewis and B. H. Booms. **The Marketing Aspects of Service Quality**. In *Emerging Perspectives in Services Marketing*. American Marketing Association, 1983. 28

[97] A. Parasuraman, Valarie A. Zeithaml, and Leonard L. Berry. **A Conceptual Model of Service Quality and Its Implications for Future Research**. *The Journal of Marketing*, **49**(4):pp. 41–50, 1985. 28

[98] Tigineh Mersha and Veena Adlakha. **Attributes of Service Quality: The Consumers' Perspective**. *International Journal of Service Industry Management*, **3**, 1992. 28

[99] A Parasuraman, Valarie A Zeithaml, and Leonard L Berry. **SERVQUAL: A Multiple-Item Scale For Measuring Consumer Perception**. *Journal of Retailing*, **64**(1):12–40, 1988. 28

[100] Leyland F. Pitt, Richard T. Watson, and C. Bruce Kavan. **Service Quality: A Measure of Information System Effectiveness**. *Management Information Systems Quarterly*, **19**, 1995. 28

[101] Graham D April and Shaun Pather. **Evaluating Service Quality Dimensions within e-Commerce SMEs.** *Electronic Journal of Information Systems Evaluation*, **11**(3):109–124, 2008. 28

[102] A. Sahai, S. Graupner, V. Machiraju, and A. van Moorsel. **Specifying and monitoring guarantees in commercial grids through SLA**. In *Cluster Computing and the Grid, 2003. Proceedings. CCGrid 2003. 3rd IEEE/ACM International Symposium on*, pages 292 – 299, may 2003. 28

[103] Daniel A. Menascè and Emiliano Casalicchio. **Quality of Service Aspects and Metrics In Grid Computing**. In *Computer Measurement Group Conference*, pages 521–532, 2004. 28

[104] Pankesh Patel, Ajith Ranabahu, and Amit Sheth. **Service Level Agreement in Cloud Computing**. *Cloud Workshops at OOPSLA09*, pages 1–10, 2009. 28

[105] Linlin Wu and Rajkumar Buyya. **Service Level Agreement (SLA) in Utility Computing Systems**. *CoRR*, **abs/1010.2881**, 2010. 28, 35

[106] Alexander Keller, Er Keller, and Heiko Ludwig. **Defining and Monitoring Service Level Agreements for dynamic e-Business**. In *in Proceedings of the 16th USENIX System Administration Conference*. The USENIX Association, 2002. 28, 37

[107] Constantinos Kotsokalis and Ulrich Winkler. **Translation of Service Level Agreements: A Generic Problem Definition**. In Asit Dan, Frdric Gittler, and FaroukEditors Toumani, editors, *The 3rd Workshop on NonFunctional Properties and SLA Management in ServiceOriented Computing NFPSLAMICSOC 2009*, **6275**, pages 248–257. Springer Berlin / Heidelberg, 2010. 28

[108] Christopher Ward, Melissa J. Buco, Rong N. Chang, Laura Z. Luan, Edward So, and Chunqiang Tang. **Fresco: A Web Services based Framework for Configuring Extensible SLA Management Systems**. In *International Conference on Web Services*, pages 237–245, 2005. 28

[109] Omer F Rana, Martijn Warnier, Thomas B Quillinan, and Frances M T Brazier. **Monitoring and Reputation Mechanisms for Service Level Agreements**. *Grid Economics and Business Models GECON*, pages 125–139, 2008. 28, 36, 37, 38, 104

[110] Philip Bianco, Grace Lewis, and Paulo Merson. **Service Level Agreements in Service-Oriented Architecture Environments**. Technical report, Software Engineering Institute - Carnegie Mellon, 2008. 29

[111] Bryce Mitchell and Paul Mckee. **SLAs A Key Commercial Tool**. 29

[112] Heiko Ludwig, Alexander Keller, Asit Dan, Richard P King, and Richard Franck. **Web service level agreement (WSLA) language specification**. *IBM Corporation*, pages 815–824, 2003. 33

[113] Alexander Keller and Heiko Ludwig. **The WSLA framework: Specifying and monitoring service level agreements for web services**. *Journal of Network and Systems Management*, **11**(1):57–81, 2003. 33, 36

[114] Stuart E. Middleton, Mike Surridge, Siegfried Benkner, and Gerhard Engelbrecht. **Quality of Service Negotiation for Commercial Medical Grid Services**. *Journal of Grid Computing*, **5**:429–447, 2007. 33

[115] Avraham Leff, James T Rayfield, and Daniel Dias. **Meeting service level agreements in a commercial grid**. *IEEE Internet Computing*, **7**:44–50, 2003. 33

[116] Surya Nepal, John Zic, and Shiping Chen. **WSLA+: Web Service Level Agreement Language for Collaborations**. In *IEEE International Conference on Services Computing*, pages 485–488, 2008. 33

[117] A Andrieux, K Czajkowski, A Dan, K Keahey, H Ludwig, J Pruyne, J Rofrano, S Tuecke, and M Xu. *Web services agreement specification (WS-Agreement)*, pages 1–47. Number GFD.107 in . The Global Grid Forum (GGF), 2004. 33

[118] Dominic Battré, Odej Kao, and Kerstin Voss. **Implementing WS-Agreement in a Globus Toolkit 4.0 Environment**. In *Grid Middleware and Services*, pages 409–418. Springer, 2008. 34

[119] Pankesh Patel, Ajith Ranabahu, and Amit Sheth. **Service level agreement in cloud computing**. 2009. 34

[120] Vladimir Tosic, Hanan Lutfiyya, and Yazhe Tang. **Web Service Offerings Language (WSOL) Support for Context Management of Mobile/Embedded XML Web Services**. In *Advanced Industrial Conference on Telecommunications*, 2006. 34

[121] D.D. Lamanna, J. Skene, and W. Emmerich. **SLAng: a language for defining service level agreements**. In *Distributed Computing Systems, 2003. FTDCS 2003. Proceedings. The Ninth IEEE Workshop on Future Trends of*, pages 100 – 106, may 2003. 34

[122] A. Kotok and D. R. R. Webber. **EbXML: The New Global Standard for Doing Business Over the Internet**. 2002. 35

[123] Francisco Curbera, Yaron Goland, Johannes Klein, Frank Leymann, S Weerawarana, et al. **Business process execution language for web services, version 1.1**. 2003. 35

[124] David Ameller and Xavier Franch. **Service Level Agreement Monitor (SALMon)**. In *International Conference on COTS-Based Software Systems*, pages 224–227, 2008. 36

[125] Anton Michlmayr, Florian Rosenberg, Philipp Leitner, and Schahram Dustdar. **Comprehensive QoS monitoring of Web services and event-based SLA violation detection**. In *Proceedings of the 4th International Workshop on Middleware for Service Oriented Computing*, MWSOC '09, pages 1–6, New York, NY, USA, 2009. ACM. 37

[126] **Dictionary.com Unabridged**. May 2012. 38

[127] Ronald R. Yager. **A Game-theoretic Approach to Decision Making Under Uncertainty**. *International Journal of Intelligent Systems in Accounting, Finance & Management*, **8**(2):131–143, June 1999. 38

[128] George J. Klir. *Uncertainty and Information: Foundations of Generalized Information Theory*. Wiley-IEEE Press, 2005. 39

[129] R Kacker and A Jones. **On use of Bayesian Statistics to make the Guide to the Expression of Uncertainty in Measurement consistent**. *Metrologia*, **40**(5):235–248, October 2003. 39

[130] Frank H. Knight. *Risk, Uncertainty, and Profit*. Boston, MA: Hart, Schaffner & Marx; Houghton Mifflin Co., 1921. 39, 48, 55

[131] Millett G. Morgan and Max Henrion. *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press, New York, August 1990. 39

[132] Eoghan Casey. **Error, Uncertainty and Loss in Digital Evidence**. *IJDE*, **1**(2), 2002. 40

[133] P. Busch, T. Heinonen, and P. Lahti. **Heisenberg's uncertainty principle**. *Physics Reports*, **452**(6):155–176, November 2007. 40

[134] T. N. Palmer. **Predicting uncertainty in forecasts of weather and climate**. *Reports on Progress in Physics*, **63**(2):71–116, 2000. 40

[135] Joop Koppenjan and Erik-Hans Klijn. *Managing uncertainties in networks : a network approach to problem solving and decision making*. Routledge, 2004. 40

[136] M. B. Beck. **Water quality modeling: A review of the analysis of uncertainty**. *Water Resources Research*, **23**(8), 1987. 41

[137] Stephen C. Hora. **Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management**. *Reliability Engineering and System Safety*, **54**(2-3):217 – 223, 1996. 41, 51

[138] Armen Der Kiureghian and Ove Ditlevsen. **Aleatory or epistemic? Does it matter?** *Structural Safety*, **31**(2):105–112, March 2009. 41, 51

[139] William L. Oberkampf, Jon C. Helton, Cliff A. Joslyn, Steven F. Wojtkiewicz, and Scott Ferson. **Challenge problems: uncertainty in system response given uncertain parameters**. *Reliability Engineering & System Safety*, **85**(1-3):11 – 19, 2004. 41, 51

[140] E. Zio and G.E. Apostolakis. **Two methods for the structured assessment of model uncertainty by experts in performance assessments of radioactive waste repositories**. *Reliability Engineering & System Safety*, **54**(2-3):225 – 241, 1996. 41, 52

[141] P. John Clarkson and Claudia Eckert, editors. *Design Process Improvement: A review of current practice*. Springer, 1st edition. edition, December 2004. 42, 51

[142] Melanie Kreye, Yee-Mey Goh, and Linda Newnes. **Manifestation of uncertainty - A classification**. In *International Conference on Engineering Design, ICED11*, pages 1–12, August 2011. 42, 54

[143] W. Walker, P. Harremoës, J. Rotmans, P. J. van der Sluijs, M. van Asselt, P. Janssen, and M. P. K. von Krauss. **Defining uncertainty: A conceptual basis for uncertainty management in model-based decision support**. *Integrated Assessment*, **4**:5–17, 2003. 42, 44, 45, 51, 70

[144] William C. Howell and Sarah A. Burnett. **Uncertainty measurement: A cognitive taxonomy**. *Organizational Behavior and Human Performance*, **22**(1):45 – 68, 1978. 43, 51

[145] Daniel Kahneman and Amos Tversky. **Variants of uncertainty**. *Cognition*, **11**(2):143 – 157, 1982. 43, 51

[146] Olivier De Weck, Claudia Eckert, Olivier De Weck, and Claudia Eckert. **A Classification of Uncertainty for Early Product and System Design**, 2007. 44, 51

[147] Mordecai Kurz. **Rational beliefs and endogenous uncertainty.** *Economic Theory*, **8**(3):383, 1996. 44

[148] Mordecai Kurz. **Endogenous Uncertainty: A Unified View of Market Volatility**. Technical Report 98013, Stanford University, Department of Economics, 1998. 44

[149] John P. Norton, James D. Brown, and Jaroslav Mysiak. **To what extent, and how, might uncertainty be defined? Comments engendered by 'Defining uncertainty: a conceptual basis for uncertainty management in model-based decision support': Walker et al., Integrated Assessment 4:1, 2003**. *Integrated Assessment Journal*, **6**:83–88, 2006. 45

[150] Merle H. Mishel. **Perceived uncertainty and stress in illness**. *Research in Nursing and Health*, **7**(3):163–71, 1984. 45, 46, 52

[151] Merle H. Mishel. **Uncertainty in Illness**. *Journal of Nursing Scholarship*, **20**(4):225–232, 1988. 45, 46, 52

[152] Austin S. Babrow, Chris R. Kasch, and Leigh A. Ford. **The Many Meanings of Uncertainty in Illness: Toward a Systematic Accounting**. *Health Communication*, **10**(1):1–23, 1998. 45, 46, 52

[153] Jürgen Kasper, Friedemann Geiger, Saskia Freiberger, and Angelika Schmidt. **Decision-related uncertainties perceived by people with cancer modelling the subject of shared decision making**. *Psycho-Oncology*, **17**(1):42–48, 2008. 45, 46, 53

[154] Richard L. Priem, Leonard G. Love, and Margaret A. Shaffer. **Executives Perceptions of Uncertainty Sources: A Numerical Taxonomy and Underlying Dimensions**. *Journal of Management*, **28**(6):725–746, 2002. 48

[155] Andrew W Lo and Mark T Mueller. **WARNING: Physics Envy May Be Hazardous To Your Wealth!** *Group*, **8**(2):1363, 2010. 48, 49, 54

[156] Willem Schaafsma. **The Neyman-Pearson theory for testing statistical hypotheses**. *Statistica Neerlandica*, **25**(1):1–27, 1971. 55

[157] E. L. Lehmann and Joseph P. Romano. *Testing statistical hypotheses*. Springer Texts in Statistics. Springer, New York, third edition, 2005. 55

[158] K. Durga Rao, H.S. Kushwaha, A.K. Verma, and A. Srividya. **Quantification of epistemic and aleatory uncertainties in level-1 probabilistic safety assessment studies**. *Reliability Engineering & System Safety*, **92**(7):947 – 956, 2007. 55

[159] L.A. Zadeh. **Fuzzy sets**. *Information and Control*, **8**(3):338 – 353, 1965. 55

[160] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976. 55

[161] James O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer, 2nd edition, 1985. 55

[162] Ramon E. Moore. *Methods and Applications of Interval Analysis*. Society for Industrial Mathematics, 1987. 55

[163] Douglas W. Hubbard. *How to Measure Anything: Finding the Value of "Intangibles" in Business*. John Wiley & Sons, 2007. 55

[164] Stephen Paul Marsh. **Formalising trust as a computational concept**. Technical report, , 1994. 57

[165] Anil Kini and Joobin Choobineh. **Trust in Electronic Commerce: Definition and Theoretical Considerations**. *Hawaii International Conference on System Sciences*, **4**:0051, 1998. 57, 58

[166] D. Harrison Mcknight and Norman L. Chervany. **What Trust Means in E-Commerce Customer Relationships: An Interdisciplinary Conceptual Typology**. *International Journal of Electronic Commerce*, **6**:35–59, 2002. 57

[167] Audun Jøsang and Stéphane Lo Presti. *Analysing the Relationship between Risk and Trust*, **2995**, pages 135–145. Springer Berlin / Heidelberg, 2004. 57, 58

[168] Bjørnar Solhaug, Dag Elgesem, and Ketil Stølen. **Why trust is not proportional to risk**. In *Proceedings of the 2nd International Conference on Availability, Reliability and Security (ARES07)*, pages 11–18, 2007. 58

[169] Bjørnar Solhaug and Ketil Stølen. **Uncertainty, subjectivity, trust and risk: how it all fits together**. In *Security and Trust Management*, pages 1–5. Springer, 2012. 58

[170] Anna Durante, David Bell, Louis Goldstein, Jon Gustafson, and Harumi Kuno. **A model for the E-service marketplace**. *HP LABORATORIES TECHNICAL REPORT HPL*, (17), 2000. 63

[171] T. Rajendran and P. Balasubramanie. **Analysis on the Study of QoS-Aware Web Services Discovery**. *Computing Research Repository*, **abs/0912.3**, 2009. 65

[172] Elisabetta Di Nitto, Massimiliano Di Penta, Alessio Gambi, Gianluca Ripa, and Maria Luisa Villani. *Negotiation of Service Level Agreements: An Architecture and a Search-Based Approach*. 2007. 74

[173] Sebastian Hudert, Heiko Ludwig, and Guido Wirtz. **Negotiating SLAs-An Approach for a Generic Negotiation Framework for WS-Agreement**. *Journal of Grid Computing*, **7**:225–246, 2009. 74

[174] James Skene, Allan Skene, Jason Crampton, and Wolfgang Emmerich. **The monitorability of service-level agreements for application-service provision**. In *Workshop on Software and Performance*, pages 3–14, 2007. 75

[175] Jonathan Vogel. **Epistemic Bootstrapping**. *Journal of Philosophy*, **105**(9):518–539, 2008. 78

[176] Andr Gabor and C. W. J. Granger. **Price as an Indicator of Quality: Report on an Enquiry**. *Management Decision*, **17**:590–618, 1979. 79

[177] Min Ding, William T. Ross Jr., and Vithala R. Rao. **Price as an Indicator of Quality: Implications for Utility and Demand Functions**. *Journal of Retailing*, **86**:69–84, 2010. 79

[178] Jørgen Drud Hansen and Jørgen Ulff-Møller Nielsen. **Price as Indicator for Quality?** 2007. 79

[179] Algirdas Avizienis, Jean claude Laprie, Brian Randell, and Carl E. Landwehr. **Basic Concepts and Taxonomy of Dependable and Secure Computing**. *IEEE Transactions on Dependable and Secure Computing*, **1**:11–33, 2004. 82

[180] Algirdas Avizienis. **Fault-Tolerant Systems**. *IEEE Transactions on Computers*, **25**:1304–1312, 1976. 82

[181] Caroline J Wesson. *The Communication and Influence of Confidence and Uncertainty*. PhD thesis, University of Wolverhampton, Nov 2005. 83

[182] Dane K. Peterson and Gordon F. Pitz. **Confidence, uncertainty, and the use of information**. *Journal of Experimental Psychology: Learning, Memory, and Cognition.*, **14**(1):85–92, 1988. 83

[183] Colin Howson and Peter Urbach. *Scientific Reasoning: The Bayesian Approach*. Open Court Publishing Company, 1993. 89

[184] M. Bayes and M. Price. **An Essay towards Solving a Problem in the Doctrine of Chances. By the Late Rev. Mr. Bayes, communicated by Mr. Price, in a letter to John Canton, M. A. and F. R. S.** *Philosophical Transactions (1683-1775)*, 1763. 89

[185] D. R. Bellhouse. **The Reverend Thomas Bayes, FRS: A Biography to Celebrate the Tercentenary of His Birth**. 2001. 89

[186] S. A. McLeod. **Simply Psychology; Likert Scale**, 2008. Online; Accessed 22-May-2012. 92

[187] A. Bowling. **Research Methods in Health: Investigating Health and Health Services**. 2002. 92

[188] George J. Stigler. **The Development of Utility Theory. I**. *Journal of Political Economy*, **58**, 1950. 95

[189] P. Samuelson. **Foundations of economic analysis**. 1967. 95

[190] Walter Nicholson. **Microeconomic Theory: Basic Principles and Extensions**. 1995. 95

[191] Maria Eugnia Mata. **Cardinal Versus Ordinal Utility: Antnio Horta Osrio's Contribution**. *Journal of The History of Economic Thought*, **29**, 2007. 95

[192] Peter Fishburn. **A variational model of preference under uncertainty**. *Journal of Risk and Uncertainty*, **8**:127–152, 1994. 96

[193] Robert P. King and Lindon J. Robison. **An Interval Approach to Measuring Decision Maker Preferences**. *American Journal of Agricultural Economics*, **63**, 1981. 96

[194] Hans P. Binswanger. **Attitudes toward Risk: Experimental Measurement in Rural India**. *American Journal of Agricultural Economics*, **62**, 1980. 96

[195] Urszula Chajewska, Daphne Koller, and Dirk Ormoneit. **Learning an Agent's Utility Function by Observing Behavior**. In *International Conference on Machine Learning*, pages 35–42, 2001. 96

[196] Oskar Morgenstern and John Von Neumann. *Theory of Games and Economic Behavior*. Princeton University Press, 1944. 97

[197] Ward Edwards. **The theory of decision making.** *Psychological Bulletin*, **51**(4):380–417, July 1954. 97

[198] P.C. Fishburn. *The Foundations of Expected Utility*. D. Reidel Publishing, 1982. 97

[199] P.C. Fishburn. *Utility Theory for Decision Making*. Wiley, New York, 1970. 97

[200] H. Raiffa. **Decision analysis: introductory lectures on choices under uncertainty**. 1968. 98

[201] Jorge Cardoso, A. P. Sheth, J. A. Miller, Jonathan Arnold, and Krys Kochut. **Quality of service for workflows and web service processes**. *Journal of Web Semantics*, **1**:281–308, 2004. 98

[202] Krys J. Kochut, Amit P. Sheth, and John A. Miller. **ORBWork: A CORBA-Based Fully Distributed, Scalable and Dynamic Workflow Enactment Service for METEOR**. 1998. 98

[203] Wikibooks. **Computer Programming/Procedures and functions**, March 2011. Online; Accessed 21-April-2012. 103

[204] Wikiversity. **Introduction to Programming/Subprograms**, November 2011. Online; Accessed 21-April-2012. 103

[205] Sameer Ajmani, Robert Morris, and Barbara Liskov. **A Trusted Third-Party Computation Service**. Technical Report MIT-LCS-TR-847, MIT, May 2001. 105

[206] Kathryn M. Kimery and Mary Mccord. **Third-party assurances: the road to trust in online retailing**. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on*, page 10 pp., jan. 2002. 105

[207] P J Skevington and T P Hart. **Trusted third parties in electronic commerce**. *BT Technology Journal*, **15**:39–44, 1997. 10.1023/A:1018628522847. 105

[208] Diomidis Spinellis, Stefanos Gritzalis, John Iliadis, Dimitris Gritzalis, and Sokratis Katsikas. **Trusted third party services for deploying secure telemedical applications over the WWW**. *Computers & Security*, **18**(7):627 – 639, 1999. 105

[209] Ilia Tsetlin. **A Method for Eliciting Utilities and its Application to Collective Choice**. *Theory and Decision*, **61**:51–62, 2006. 10.1007/s11238-006-7191-7. 111

[210] J. C. Hershey and P. J. H. Schoemaker. **Probability Versus Certainty Equivalence Methods in Utility Measurement: Are They Equivalent**. 1985. 112