Trust and Reputation in Web Services

A thesis presented
by

Abul Kalam Rafi Ahmed

to
The Department of Computer Science
in partial fulfillment of the requirements
for the degree of
Master of Science
in the subject of

Computer Science

The University of Manitoba
Winnipeg, Manitoba

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES
*****
COPYRIGHT PERMISSION

Trust and Reputation in Web Services

BY

Abul Kalam Rafi Ahmed

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

MASTER OF SCIENCE

Thesis advisor

Dr. Vojislav B. Mišić

Author

Abul Kalam Rafi Ahmed

Trust and Reputation in Web Services

Abstract

Web Services are service-oriented, asynchronous and loosely coupled message-passing protocols that enable communicating heterogeneous applications over the Internet. However, current standards of Web Services provide just the basic functionality, especially with respect to the selection of the provider among several alternative choices. In the current approach, the information about possible providers is so generic that it becomes very difficult for the client to select the most suitable one. To overcome this shortcoming, we propose to introduce the concepts of trust and reputation in the selection process. In this approach, previous experiences and feedback from other clients will be utilized in the process of selecting the best provider for a particular task. To this end, we describe 'Dynamic Trust and Reputation Management' (DTRM), a suitable trust and reputation mechanism that categorizes the feedback providers and assigns appropriate weights to their ratings. The mechanism also discusses the implementation options in a centralized as well as a distributed, peer-to-peer environment. The proposed mechanism is verified through simulations.

Contents

List of Figures

List of Tables

Acknowledgments

In the name of Allah, the merciful, the compassionate.

I would like to express my gratitude to my research supervisor Dr. Vojislav B. Mišić, whose amazing patience, infinite help in all aspects, and non-stop support made me achieve what I did not know I had in me. This thesis would not have been possible without his guidance and help. I would also like to express my special thanks to my thesis committee members.

I would like to thank the Department of Computer Science, for their support and the Graduate Studies Committee of the Department of Computer Science for their constructive comments and suggestions on my thesis proposal.

I thank my friends Mostafizur Rahman, Subrata Shaha and Faisal Shibley for their suggestions while I needed.

Finally, I wish to thank my wife for her continuous support and encouragement, which gave me the strength to complete this thesis.

Abul Kalam Rafi Ahmed

University of Manitoba

May 2007

This thesis is dedicated to my parents Mrs. Rabeya Gafur and Mr. Abdul Gafur.

Chapter 1

Introduction

Web Services have become an essential part of today's electronic business. Web Services are a collection of technologies that aim to enable and facilitate interaction of heterogeneous business applications over the Internet [6]. The interaction is service-oriented, *i.e.,* some applications request the services of others, and asynchronous, *i.e.,* interaction uses loosely coupled message-passing protocols. Normally, such interactions should be performed among business applications only, without human intervention. In this manner, business applications can collaborate regardless of physical and geographical distance or the underlying hardware, software, and application platform. This should lead to reductions of cost, as well as to improvements in functionality, timeliness, and overall quality of service.

1.1 Motivation

Web Services are a standard way of interoperating between different software applications which are running on different platforms. In Web Services, programs interact with other programs to find a suitable service dynamically. This interaction takes place using a standard messaging language - Extensible Markup Language (XML). All these can be combined in a loosely coupled way to smooth the progress of online transaction [4].

Web Services use open standards such as Web Service Description Language (WSDL) for service description, Universal Description, Discovery and Integration (UDDI) for service discovery, and Simple Object Access Protocol (SOAP) for communication between services [6]. All of those protocols are based on the Extended Markup Language (XML), which ensures the widest possible acceptance and ease of deployment of systems that use them. Three main roles have been defined in the Web Services architecture: requester (which becomes a service client in future), service provider, and registry.

- The provider publishes the service description to the service registry;
- The client queries the registry to find suitable services;
- The client then selects the most suitable one and then negotiates with its provider in order to bind the service and actually use it.

Figure 1.1 illustrates this scenario.

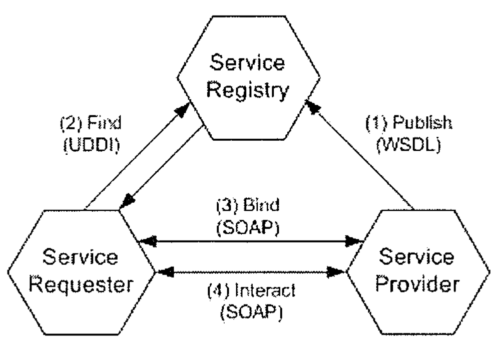

Figure 1.1: Basic Web Services roles

While Web Services hold great promise, current standards support only basic interoperability. In particular, the process of selecting the most appropriate (*i.e.,* the best) service for the desired operation is not well supported, since the WSDL description of a Web Service provides just the basic information which is necessary to use the service. Any extended information related to the quality guarantees of the service, as well as the credentials of the service provider, are nonexistent. These are the areas which are vital but not yet taken into consideration.

1.2 Research Goal

Web Services are considered to be the future platform of internet business [31]. Service selection is an important functionality of Web Services that has yet to be improved; in particular, one of the main motivations for introducing Web Services - the ability to select an optimum service dynamically - is not fully supported by the current infrastructure standards. To address these crucial issues, we propose a novel trust- and reputation-based mechanism which can assist in selecting the best possible service dynamically.

The objective of this thesis is to:

- describe the mechanism for trust and reputation and support which is necessary to use trust and reputation in the selection process, and
- evaluate the behavior and performance of such a scheme through simulation.

1.3 Proposed Solution

Inadequate information about a service makes it hard for a requester to select the right provider. One way to overcome this issue is to make the decision according to the trust in the service provider [9]; other criteria such as price or performance promises can be used as well. Trust can be defined as *"willingness to commit to a collaborative effort before you know how the other person will behave"* [10]. In this case, trusting a provider means that we believe it will fulfill its promises as advertised. If the requester has had previous interactions with a particular provider, the information about the experiences gained (be they positive satisfactory or not) can be utilized to decide whether to trust the provider (and engage in another interaction) or not.

In many cases there has been no previous interaction and, hence, there is no trust information available. A possible solution is to rely on the experiences of others, and utilize somehow the trust information they have, instead of our own [20]. This is often referred to as reputation, which can be informally defined as *"a collection of perception and beliefs, both past and present, which measures to lower the uncertainty about a possible action"* [23]. In other words, when interacting with 'strangers', one has to use every bit of information available, and reputation is one such source of information [8]. As with trust, it is assumed that the likelihood of obtaining better service is higher when the provider has a better reputation: in other words, reputation is a kind of 'indirect trust' based on the feedback from other sources. From the provider standpoint, achieving better reputation will improve the likelihood of doing business and, ultimately, leads to increased revenue [32].

Of course, we can use other clients' experience even if we have those of our own, *i.e.,* we can combine trust and reputation, if available, in order to select the best provider for a given service.

Furthermore, recommendation is very crucial in a place like Web Services where one needs to believe another in certain common issues for the sake of interaction. However, it is not fair that all the recommendations or feedback from other sources are scaled equally. Now, the recommenders are classified into groups according to their trustworthiness to ensure credible feedback. Another way to obtain a credible feedback is to keep track of previous transactions of the providers. In both the cases these will increase significant amount of work load. The service registry alone will face difficulty to handle all these points mentioned above. Even if it is possible, this will make the registry slow and thus less effective.

While trust and reputation are widely used in everyday life, so far only a handful of authors have considered using those mechanisms to enrich the Web Services paradigm. In this thesis, we plan to investigate the implementation of a suitable trust- and reputation- based mechanism to help in the selection phase of Web Service interaction.

To address the problems of dynamic selection of service providers, two approaches: distributed and centralized can be utilized. For the distributed approach, the peer to peer model can be used, in which each client will keep track of its experiences and share them upon request. For the centralized approach, we propose to use a dedicated 'Transaction Registry', which will keep track of all the transactions and stand behind to check whether all the promises made by the respected providers are fully maintained. In the following, we will describe those in some detail.

In both cases, the same Dynamic Trust and Reputation Management (DTRM for short) model for Web Services is proposed. The DTRM Model will calculate the trust and reputation of the given provider based on a client's own experiences as well as the opinions from different groups of the society.

The proposed solution is not limited to Web Services only; the trust and reputation mechanism can easily be applied to most, if not all, service oriented environments.

1.4 Thesis Contribution

In this thesis, we simplify the selection process for the clients.

- we propose to add trust and reputation to Web Services
- we define a novel trust and reputation model that takes into account experiences of all clients, in a socially stratified manner
- we evaluate its performance and show that it allows for more precise choice of providers than a simple randomized approach which is the only solution available so far.

1.5 Thesis Organization

The thesis starts with the theoretical overview on Web Services, its structure and the basic concept of trust and reputation, followed by the literature review where we described the current issues of trust and reputation in both electronic marketplace and Web Services. At the end of this chapter a comparison of the previous proposed model is drawn.

In Chapter 4 the DTRM Model is explained thoroughly. In this chapter the two different approaches and their corresponding work-flows are discussed. Here the implementation process of DTRM is also talked about.

Chapter 5 presents some of the results we obtained implementing the DTRM in the simulator we built. Before summarizing the results in the next chapter, we also analyzed the results noting our contribution in the process.

In the last chapter we discuss the possible future extension of the thesis. At the end of the thesis a conclusion is made with the research findings.

Chapter 2

Related literature and theoretical focus

The significance of Web Services is established; however, there are some limitations of Web Services. In this chapter, we are going to explicate Web Services behavior, their structure along with the general concept of trust and reputation.

2.1 Web Services

Web Services are loosely coupled, platform-independent standard that allows short-term co-operations between services based on XML [14].

According to [28],

> *"A Web Service is a software system identified by a URL, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web Service in a manner prescribed by its definition, using XML-based messages conveyed by internet protocols."*

Web Services offer many benefits but also suffer from some disadvantages. Here is a short discussion about the advantages and limitations of Web Services.

There are a number of technical and business benefits for which the Web Services became well accepted throughout the world of e-business. Web Services use open, XML based standards that makes the mechanism easy to utilize. Use of Web Services enables applications to talk to each other through a standard approach. The interoperability mechanism of Web Services make communication simpler and faster. Web Services are easy to implement and this implementation can be done with an incremental approach. Besides, in the same Web Services infrastructure, services and components can be reused. Furthermore, Web Services are loosely coupled and thus it facilitates distributed communication. All the above mentioned benefits can significantly reduce the development and operational costs of Web Services [8; 31].

Web Services suffer from the lack of a standard security procedure. Therefore, many companies are not yet interested as these might need further development. Web services also did not achieve full confidence of clients and providers. QoS support of Web Services is still not up to the mark in terms of defining and guaranteeing adequate service. Web Services are still to answer whether the provider can meet the expectation, whether the process time will be short enough, whether the updates will be frequent, whether it can handle a vast number of requests etc. Web Services may cause troubles like incomplete transaction, and revenue loss that will instill lack of confidence on the part of users [8; 31].

2.2 Web Services Structure

To discuss the Web Services structure and to understand the inside mechanism, we need to review its main standards: WSDL, UDDI, and SOAP. In this section, we are going to focus on the main functionality of those standards.

2.2.1 WSDL

The Web Service Description Language (WSDL) is an XML schema format that defines and extensible framework for describing Web Services interfaces. The WSDL describes the actions of Web Services as a contract between the client and server. Using it they can exchange data in a standard way. WSDL was originally created by IBM and Microsoft, merging three proposals: IBM's Network Accessible Service Specification Language (NASSL) along with Microsoft's SOAP Contact Language (SCL) and Service Description Language (SDL) [15].

A registry stores the URLs of WSDL documents. Registries store files which provide specific technical information. Using this data, applications can query and communicate with Web Services. The client application uses WSDL, an XML based language to obtain such information [14]. Each message of WSDL is bound to a concrete network protocol (HTTP, SOAP) and message format. Thus, a message consists of a collection of typed data items such as character string, date and time. An exchange of messages between the entities, as in a transaction, is considered as an operation. A WSDL operation has a set of pre-defined messages; similarly, a set of operations is called a port type. WSDL is used with SOAP, HTTP and MIME. It also describes where the service can be found and how it should be invoked [15].

The goal of WSDL is to ensure that automated processes will determine the remote systems without human intervention or prior definitions [13]. The WSDL document specifies the services capabilities, its location on the web and instructions regarding how to access it. A WSDL document defines the structure of the message that Web Services send and receive. Using this information, applications searching for Web Services to fill a specific need can analyze the WSDL files of several comparable services and choose between the services. WSDL specifies how to structure documents so that Web Services communication can be automated. Each WSDL document contains XML elements that define the characteristics and capabilities of a Web Service. The main element belongs to one of the two categories shown in Figure 2.1 [15].:

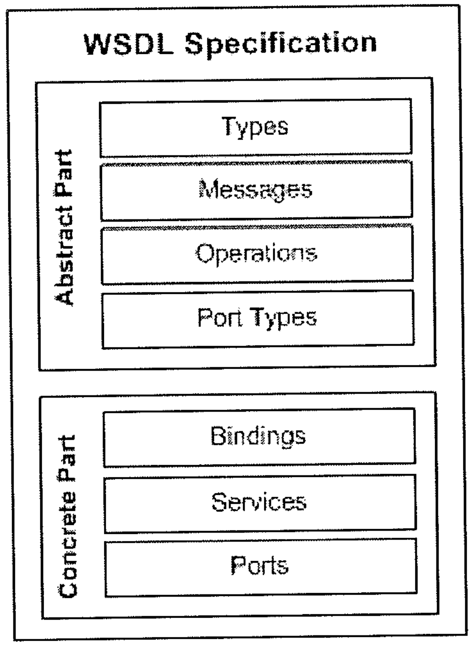

Figure 2.1: WSDL specification

- abstract definitions: defines general concepts which can apply to more than one instance. Types, messages, operations and port types are considered as the abstract definitions.
- concrete definitions: defines specific examples that apply to real interactions. Binding, port, service are considered as concrete definitions.

2.2.2 UDDI

Universal Description, Discover and Integration (UDDI) is one of the major standards of Web Services. *"UDDI is a XML based, platform-independent specification for a registry of information for Web Services."* [11]. It was designed by Microsoft, IBM, and Ariba and initially published in September 2000. The latest UDDI version 3.0.2 was realized in 2005 [5].

UDDI enables organizations to make a standard way of organizing businesses, discovering, reusing and managing Web Services. It can be a very powerful resource of organizing Web

Services at design time and at run-time it provides dynamic binding information. UDDI describes the design of the registry and the procedure of describing, publishing and discovering services [22].

The provided information of a provider can be stored in a private UDDI registry accessible by assigned business partners or in a public UDDI registry accessible by any interested clients. UDDI registry also provides binding information dynamically at run time [5]. The structure of a UDDI registry is similar to a phone book. The providers can describe their own services along with discovering same kind of services in other companies and integrate all in the system through UDDI. Clients need to search about services through UDDI.

The main tasks of UDDI [5] are:

- Publish Web Services information and categorize rules,
- Find descriptions of the requested Web Services,
- Discover information about service providers,
- Drive better code reuse and developer productivity,
- Determine necessary protocol and parameters to invoke services.

2.2.3 SOAP

Web Services depend on Simple Object Access Protocol (SOAP) for communication between entities regardless of their programming environment and operating system. SOAP is an XML-based communication protocol [30] developed by Userland software, DeveloperMentor, and Microsoft and published its first version in 1999.

SOAP meets its goal of eliminating the barriers of heterogeneity by following web protocol which is simple, flexible, firewall friendly and XML based. It establishes a set of rules which enable both the clients and the servers to invoke remote procedure using SOAP as a communications framework [22].

Web Services applications communicate among themselves through SOAP messagea. A SOAP message needs to follow a particular format so that others can interpret the message easily and correctly. A SOAP message has three parts: envelope, header and body as shown in Figure 2.2. The application specific data is kept in the body whereas the optional part header contains issues like security. The envelope contains these two parts and wraps the whole message. These messages are interpreted and received by SOAP server [15].

SOAP offers the following features [25]:

- SOAP exchanges information or messages between two services in a very structured and typed way.
- It is a platform independent mechanism that reduces the cost and complexity of application integration.
- Exchanging message using SOAP allows wide range of distributed approaches. Thus the services can communicate using any of the message exchange patterns.
- Generally, the SOAP messages are sent out in one direction from sender to receiver. However, another messaging is also possible that will combine multiple one-way messages.

The basic functionality of Web Services, as described with WSDL, UDDI, and SOAP, lacks some important features. At the implementation level the users need to provide serves with more accuracy and security. So far WSDL, UDDI and SOAP structure have not considered the trust and reputation concept though this is important for electronic businesses [20]. The trust and reputation concept is applied in areas such as electronic marketplace successfully. For this the concept can be implemented is Web Services.

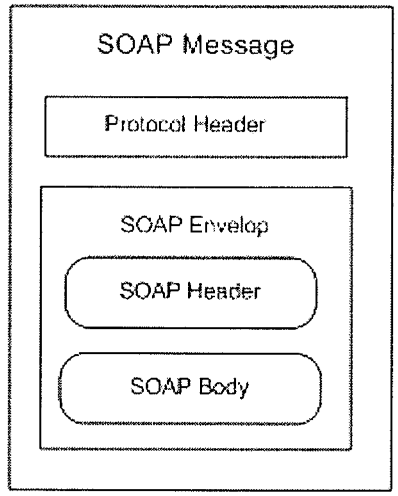

Figure 2.2: SOAP Message format

2.3 Trust and Reputation

Today's electronic market place has to confront a number of challenges such as inadequate security, lack of quality of service and cheating in business. If sufficient steps are not taken to address those challenges then the confidence of clients in electronic business will go down. Therefore, the clients may refuse to involve themselves in online activities. In general, the clients have concern about the lack of security of the whole process and acquiring guaranteed services as advertised. Without handling this properly the faith of the clients cannot be grown; which will be costly for the whole industry [7].

We need to work on building the confidence of the clients. A lot of research work [8; 17; 24; 27] has been done to recover a client's confidence and make the electronic business secure, trustworthy and beneficial for the client. Building the confidence of the clients is one of the motivations of Web Services along with some other facilities like inter-operability and less human interaction. To make the functions of Web Services more effective and usable Web Services need to emphasize security, structure and service. Using the trust and reputation concept, Web Services can perform better than before.

2.3.1 Trust

Untrustworthy and unsafe electronic transactions are not impossible in this age's of internet technology, where sometimes one needs to rely on the providers without knowing much about them. The providers have to establish confidence among the clients by providing desired services, which in this thesis we define as 'Trust'. In this thesis, trust reflects the believing level of the client about the provider that the provider will provide services as advertised. We believe that the higher the trust is the higher the chance a transaction will be successful.

The trust of the client about the provider starts building when the client is satisfied with the process and flow of business of the provider. It also increases based on the quality of provided service. If a client trusts a provider that implies that the next transaction among them will be a safe and secure transaction. Though one might think security can ensure trust, that is not true for all cases. For example, if there is a hungry tiger walking around in a cage, one is safe and secure from it, but that does not mean that we trust the tiger [7].

2.3.2 Reputation

Reputation of a provider is also crucial in online business, as after analyzing reputation a client can predict how the provider could behave. On the other hand, the client may not have any prior interaction with a provider. Therefore, there is no way to know about the provider except other client's opinion (feedback) about that provider. For this thesis, the feedback of the other clients is considered as the reputation of that provider to a client. Feedback from clients are vital to be considered because it is one of the main criteria of selecting a suitable provider in our model.

Trust and reputation have been researched for the last few years in electronic business along with Web Services. New ideas and models have been proposed to develop this area by a number of researchers. In the following some models will be discussed.

2.4 Trust & Reputation in Electronic Marketplaces

As providing feedback on providers about the previous transactions is widely used in today's electronic business, a suitable starting point for our research would be why this is significant and prior work in the use of feedback concept in other areas. In this chapter, the impact of Web Services on both electronic marketplace and Web Services are discussed briefly.

Several authors have proposed to use trust and reputation concepts in electronic marketplace systems. A generic but efficient reputation system has been implemented from 1995 onwards at the popular electronic auction site eBay [3; 21]. In this system, a buyer has the option of rating the seller after trading with her. Possible ratings are positive (+1), neutral (0), or negative (-1). The total of all the ratings for the last six months is considered to be the reputation of that seller. Note that

- one buyer can assign only a single rating (positive, neutral or negative) to a single seller, regardless of the number of transactions they have had, and
- if the rating changes, the new rating will replace the old one. The introduction of the rating system has improved the popularity of the eBay trading system [21].

Another system is used by Onsale [33] in which, a brief textual comment can be added to the ratings which take values of 'positive' (4 or 5), 'neutral' (3), and 'negative' (1 or 2). Normal averaging method is used to accumulate the feedbacks. Of course, textual comments are intended to be read by humans, and are thus not very interesting for automated systems such as Web Services. In this site anyone can rate any provider without having a previous transaction [34].

Xu *et al.* [32] described a different kind of online purchase behavior model. The model

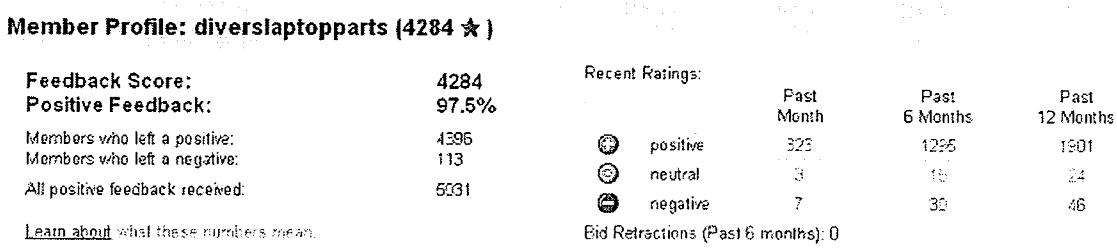

Figure 2.3: A profile of a provider of eBay [12]

contains two types of factors, product-related and seller-related. Quality and price of the product are the main concerns with respect to the product-related issue, whereas the reputation of the seller is the major concern among other issues for the buyer. Higher perceived quality and lower perceived price will have positive effect on the perceived value of a product, which will also have a positive effect on the willingness-to-buy of an online buyer. The reputation score will affect trust too; sellers with higher weight will have higher trust (which also indicates lower risk), and vice versa [32].

Many organizations use trusted third parties to help buyers to choose sellers. But when such third parties are not available, it becomes tough for a new dealer to select the proper seller. Yu and Singh [33] have proposed a social mechanism of reputation management which facilitates buying in this kind of situation, and avoids bad trading. They propose the so-called referral chain, which is the series of referrals of agents, from service requester to service provider, that had direct or indirect interaction with the provider. In this setting, if agent A wants to know the reputation of agent B, then A has to generate all possible referral chains that lead to B. Then, the reputation is calculated according to the feedback obtained through the referral chain; the calculation considers the feedback obtained from the chain, but also reputation of individual agents in the chain. This process also allows newcomer agents to quickly find reputation of the desired agent by using suitable referral chain(s) [33].

A reputation oriented reinforcement learning algorithm [26] for electronic marketplaces, based on the value and quality of individual products, has been described in [29]. The buying algorithm tries to maximize the expected value of products by selecting a reputed seller; it also dynamically maintains a set of reputed sellers to be chosen from. The selling algorithm tries to maximize the profit by altering the product quality and price, while trying to maintain (or, if possible, improve) the seller's reputation. The reputation of the sellers is dynamically calculated as well. A similar idea has been described earlier, in the form of a purchasing assistant support that will help the buyers to evaluate and rank products offered by the sellers [16].

Bolton [8] came up with a two sided reputation system, in which both the buyer and seller can rate each other. Bolton argued that it is possible to cheat the sellers as well by rating them incorrectly. This can be done intentionally or giving a fake testimony. It is found that though both way reputation system needs more information, it has better impact on trading than its single-side counterparts [8].

Azzedin [7] also mentioned the importance of trust in a network system. He points out that implementing trust is crucial for the survival of electronic business. He addressed trust as a combination of identity trust and behavior trust. He emphasized on the combination to find out a trustworthy peer. In his thesis, he compared direct trust, reputation, and direct trust and reputation for different level of dishonest domains. According to his results sometimes the direct trust and reputation works better than others and sometime the direct trust works better. However, he did not address how the trustworthy peer will affect finding a trusty peer. Moreover, feedback from all recommenders are treated with equal importance. However, that may not be true as unknown recommenders can provide false information which can affect the selection process [7].

2.5 Trust & Reputation in Web Services

While all the approaches mentioned above focus on online buying and selling products, there is no reason why any one of them could not be applied in the context of Web Services. However, this requires not only the development of suitable trust and reputation mechanisms, but also a precise description of the non-functional properties of the services offered. Some of those properties such as performance, throughput, scalability, latency, and security, are commonly referred to as Quality of Service, or QoS [19], and a number of researchers have described different aspects of QoS for Web Services. Some researchers defined QoS as a combination of availability, security, response time and throughput [19]. Others defined QoS in terms of execution price, execution duration, reputation, successful execution rate and availability [35]. In any case, service offerings from different providers can be distinguished through the provided (or perceived) QoS.

One possible way in which QoS information can be systematically recorded, maintained, and made available to potential clients is through the use of a dedicated transaction registry or monitor [27]. Tian *et al* [27] discusses the role and properties of the so-called WS-QoS broker (WSB), which keeps track of the recent information of currently available offers made by the prospective providers. It can also act as an aid in the service selection process, since it gets and tests all the required descriptions from the UDDI registry. Here 'test' means the WSB will check whether the offers can meet the requirements of the clients. It can also aid in the process of selecting the best service to invoke for the client. Furthermore, the WS-QoS monitor provides information about the available service offers [27].

Ran [24] has advocated a similar approach where an additional registry that provides certification is proposed, in addition to the regular one which provides simple directory functions. In this model, the provider first supplies the information about its service(s) to the certification registry, which checks the claims of the providers in terms of both functional and non-functional properties. After certifying that the provider's claims are true, the certifier issues an appropriate certificate to the provider. The provider then submits the usual directory information about its service to the regular UDDI registry, together with the certificate from the certification registry. When clients query the UDDI registry, they obtain both the usual directory information and the certificate. It is claimed that the use of this model helps alleviate the problems of incomplete or outdated content in the UDDI registry. It is not clear, however, how the certification registry can be implemented, and how it can verify QoS promises by a variety of providers [24].

Maximilien and Singh [18] proposed a slightly different approach where a Reputation and Endorsement system (RES) and Web Service agent proxy (WSAP) collaborate to facilitate the selection process. RES follows a set of principals which includes rating, reputation and endorsement. On the other hand, WSAP can access each service and by using it, the proxy collaborates with all the services and selects the best one suited for the client's query at hand. In addition to that, WSAP also monitors other service's activities for future usage. However, the authors did not talk about how the providers will be selected, in other words the selection criterion [18].

As current techniques do not support rational selection, Maximilien [17] proposed a new 'self-adjusting trust' based model. Self-adjusting trust is an 'autonomic' characteristic which dynamically adjusts and establishes the level of trust between the provider and requester. Each transaction is rated through automatic monitoring and user input (if available), and those ratings are used for subsequent provider selection. Emphasis is given on the way in which the service behaves as opposed in the way it advertises [17].

It should be noted that all of the approaches mentioned above require an extension of Web Services standards, as the current versions of those standards have no provisions to describe additional information such as certificates, QoS guarantees, and/or post-transaction ratings.

2.6 A Comparison of Existing Approaches

Table 2.1 compares the properties of the approaches described above. In the table, we have compared some existing models according to their ratings, referrers, selection and additional registry. As the rating can be done in different methods like cumulative, averaged or two-way, the table has three different columns for each of those. We have made two columns for whether the models have considered the credibility of the referrers and grouping of referrers. The selection process might consider QoS or dynamic selection of providers; therefore we have also made two different columns for those.

As can be seen, each of them has something to offer, however, none of them offers a comprehensive and feasible solution to the problem of introducing trust and reputation to Web Services. Here are some of the findings from the table:

- All the models do not give much importance on two-way messaging. After having a successful transaction in a two way messaging both the service requester and the service provider provides comments on each other about their experience about the recent transaction. Though it is one of the ways that can ensure fairness for both the clients and sellers.

- Only one model [3] checks the credibility of the referrer.

- All the referrers should not be scaled equally. However, classifying referrers into groups is not emphasized much.

- No more than one model considered the idea of an additional registry. For better (i.e. quick, reliable) performance the additional registry can be useful.

	ratings			referrers		selection		additional
Model	cumulative	averaged	two-way	credibility	groups	by QoS	dynamic	registry
eBay [3]	X	0	X	X	0	0	0	0
Onsale [2]	X	X	X	0	0	0	0	0
Buying and selling algorithm [29]	X	X	0	0	0	0	0	0
Extending UDDI model [24]	n/a	n/a	0	0	X	0	X	X
Two way reputation system [8]	X	0	0	0	0	0	0	0
Sporas [34]	X	X	0	0	0	0	0	0
Trust-based model [32]	n/a	n/a	X	0	0	X	0	0
Social mechanism of reputation management [33]	X	X	X	0	0	0	0	0
Service trust model [17]	X	X	X	0	0	0	0	0
Reputation and endorsement system [18]	X	X	X	0	0	0	0	0
WS-QoS framework [27]	n/a	n/a	X	0	X	0	0	0

Table 2.1: Comparison of existing approaches.

Chapter 3

Dynamic Trust and Reputation Management model for Web Services

In this thesis the Dynamic Trust and Reputation Management (DTRM) model is discussed along with implementation issues of the model. Two different approaches: centralized approach and distributed approach are elaborated after that. In the centralized approach the selection process will be done with the help of an additional transaction registry. A transaction registry will store all the previous transactions of each of the providers and check whether the promises advertised by the respected providers are fully performed or not. The detail of benefits of the new registry are explained later in this chapter. In the distributed approach the clients have to depend on the referrer for their feedback before selecting a provider. All the clients except the requester or the client who is involved in the transaction are referrers. The DTRM model will be applied in both the approaches to assist clients. More detail about the above mentioned issues along with the work flow of those are discussed in this chapter.

3.1 The DTRM model

We will now describe the proposed model and its environment. We will refer to the model as Dynamic Trust and Reputation Management, or DTRM.

3.1.1 DTRM: Calculation of Trust

The foundation of the DTRM and the main source of information about trustworthiness of a service provider is the satisfaction of the client after a service-based interaction with that particular provider. Let us call this the 'satisfaction value' and use $\sigma_{i,j}(t)$ to denote the satisfaction value of client i, for the provider j, for last t time of interactions (or, possibly, the ordinal number of the interaction). We assume that σ can take any value between 0 and 1, which correspond to totally unsatisfactory and totally satisfactory service, respectively.

Each client will maintain a record – in fact, some kind of database – about the satisfaction after service interactions. All previous interactions will be recorded there, separately for each provider. This information will be referred to as the *trust* that the current client has in the particular provider denoted by τ :

$$\tau_{i,j}(t) = \text{function of } \sigma_{i,j}(t) \text{ over all earlier } t \tag{3.1}$$

The function used can be an arithmetic mean value, which is simple but requires that the client maintains a separate record for each interaction. This is space-consuming and time-consuming, as the mean has to be recalculated for every query.

Furthermore, the simple mean assigns equal weight to all transactions regardless of when they took place; for example, the transaction performed yesterday is as important as the one from two years ago. A better solution would be to assign more weight to the more recent transactions. In this thesis, more weight is given to the k most recent (last) transactions. Here k can depend on a time frame as well. We propose to accomplish this through a suitable moving average, for example a combination of simple average and the exponentially weighted moving average (EWMA) $\overline{\tau}$ can be calculated after each transaction as

$$\overline{\tau_{i,j}(t)} = \begin{cases} \sum_{i=1}^{t} \frac{\sigma_{i,j}(i)}{t}, & 1 \leq t \leq k \\ \alpha \sum_{i=1}^{k} \frac{\sigma_{i,j}(i)}{k} + (1-\alpha)\overline{\tau_{i,j}(t-1)}, & t > k \end{cases} \tag{3.2}$$

The smoothing constant α, which takes values from 0.5 to 1, can be used to adjust the weight given to more recent transactions. A suitable value will be determined experimentally.

3.1.2 DTRM: Calculation of Reputation

As defined above, reputation of the provider from the standpoint of a particular client is simply the average satisfaction value in that provider held by other clients:

$$\rho_{i,j}(t) = \text{average of } \overline{\tau_{k,j}(t-1)} \text{ where } k \neq i \tag{3.3}$$

Averaging can take place over all other clients or only some of them, depending on whether we use a centralized or distributed model and whether the recording of satisfaction values is made mandatory or not; both of these choices will be explained in more detail below.

However, a given client may choose not to assign equal weight to the trust values of other clients. Because not all feedback is equally credible. In real life, we trust some people's opinions more than those of others: for example, opinions of family members or friends are valued higher than the opinions of unknown people. Therefore, leveling the feedbacks is crucial [33].

To account for those differences, the social classification mechanism is proposed, which is a classified client groups according to their social status. The clients are classified into three different groups: family or relatives, friends and community or others as shown in Figure 3.1.

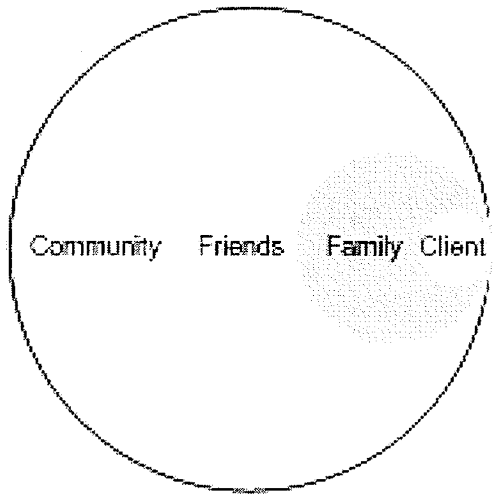

Figure 3.1: Social Classification

The feedback obtained from each of the group members is assigned decreasing weight:

$$\begin{aligned}\rho_{i,j}(t) &= w_r \cdot \text{average of } \overline{\tau_{k,j}(t-1)} \text{ from relatives} \\ &+ w_f \cdot \text{average of } \overline{\tau_{k,j}(t-1)} \text{ from friends} \\ &+ w_o \cdot \text{average of } \overline{\tau_{k,j}(t-1)} \text{ from others}\end{aligned} \tag{3.4}$$

where, for convenience, weight factors w should satisfy the following equation:

$$w_r + w_f + w_o = 1 \tag{3.5}$$

Here, the reliability of the feedback is crucial. The problem is how to assign individual clients to each of those categories. This can be accomplished in two ways. First, there can be an external channel through which such categorization is accomplished. For example, if a bank client wants to access his account online, he has to phone the bank to obtain the initial password. Similarly, some clients may be assigned the status of a relative (family member) simply because we trust them on the basis of our previous non-electronic interactions.

Second, the improvement in client's standing (and possible upgrading to a group of higher credibility) can be done by observing the agreement of reputation feedback reported by that client with the final outcome of the selection process. In other words, the more times a client recommends the service provider which gets selected and proves satisfactory in the actual interaction, the better the client's standing should be. Conversely, recommending providers that do not get selected or those that are selected but prove unsatisfactory in the actual interaction, should lead to demotion of client's status. Suitable thresholds could be established to allow for clients to move up and down.

We classify the society based on link based method and area based method which are kind of fixed for each run in the distributed approach. Therefore, in our experiments, we do not implement the improvement in client's standing rather we implemented a simplified model (DTRM) for the experiments.

3.1.3 DTRM: Implementation Issues

How can the DTRM be implemented? In the centralized approach, the DTRM can be implemented through the establishment of a Transaction Registry which may be part of the regular UDDI (directory) registry, or a separate entity. After each service transaction, the client reports its satisfaction with the service provided to this registry. Reporting may be mandatory or optional. The registry will keep track of all the transactions, and provide the information about the trust and reputation values upon request by a prospective client. In this case, clients need not keep their own records – they just send them to the Transaction registry and then forget about them. Moreover, the calculations of DTRM are performed by the registry, and the client gets all the information ready from a single source so as to help it make the decision. The sequence of message exchanges here is between the clients and Transaction registry labeled as 'Referrer'.

In the original Web Services solution, service requesters are directed to the UDDI registry; therefore, integrating the transaction and UDDI registries would speed up the querying phase and eliminate the network traffic between the two registries, as well as the risk of inconsistency between the two. To do so, the existing UDDI registry, together with UDDI and WSDL standards and the related XML schemata, needs to be augmented with appropriate capabilities. We are confident that this can be achieved in such a way as to ensure backward compatibility with the existing standards; yet such additions are beyond the scope of my thesis.

One possible drawback of the centralized approach is its fault tolerance. The entire concept falls down if the augmented registry fails for whatever reason. While this may not be a very likely event, it is still a serious problem, similar to other architectures that rely on a centralized server of any kind.

In the distributed approach, functionality of the Transaction registry is mimicked by the community of Web Service clients. Each client will have to track its own interactions with service providers and maintain accurate records of the experience of those interactions, and be able and willing to supply that information upon request. The client that needs to find a suitable service still has to query the UDDI registry to obtain a list of potential providers. Once the best candidate providers are identified, the prospective client will then issue a separate multicast (since broadcast is not practical) request to the community, asking for reputation feedback from its peers. Note that here the message exchanges between the clients and the 'Referrer', that is a community of clients and the 'Query about provider' may consist of several messages.

Note that the DTRM calculations have to be performed by each requesting client, which increases computational load of the process; but the calculations are simple and should not pose a big problem.

More importantly, the step of reputation collection could lead to high levels of network traffic, esp. if the broadcast request aims at the widest possible audience – the clients in the entire network. This traffic volume can be reduced through appropriate Time-To-Live (TTL) values which would limit the target audience of network requests to clients that can be reached with at most TTL hops away from the requesting client. Also, the reputation query message may contain a deadline beyond which the requesting client will not accept responses any more. Finally, the reputation collection process may be limited to just the few candidate providers with the highest ranking. All of these measures would help reduce network traffic. In case of no response from any of the relative, friends or community the value of reputation will be '0'. For this kind of situation the client have to depend on their trust value only.

In the distributed approach, after successfully completing each transaction, the client will share its own experience from the interaction with the community. Other clients may record this information so as to facilitate their subsequent actions. If insufficient number of responses is obtained, they can either a) repeat the request but with a longer target audience, or b) fall back on relying on its own experience only.

Regardless of the kind of architecture, security of data transmissions is an important issue. However, security is beyond the scope of this thesis, and we are assuming that issues related to security have been taken care of.

3.1.4 Hypotheses

In our thesis, we have a couple of hypothesis.

- If a group of providers is created with higher satisfaction value then those providers will be selected more often. We think that the most reliable providers group will get most of the calls.
- For a provider with higher satisfaction value the probability of getting selected will be higher than other providers. We think that the most reliable provider will get most of the calls.
- If group of clients are generated by using the social classification method then the recommendation of the family members and friends will be better than the community members. We think that different weight for each of the groups (family > friend > community) can assist the selection process best.

- If a provider has more interactions with clients then that provider will be good provider with a suitable satisfaction value. We think that the clients will suggest that provider with whom it interacts more.

3.2 Centralized Approach

For a secure business transaction, it is better to gather prior knowledge about the previous performances of a particular provider. This knowledge can be helpful while selecting a provider. Referrers can share their personal experiences. However, there is no guaranty that the experience shared among the clients will be just. To minimize that, a transaction registry is introduced. A transaction registry will keep the records of all the previous transactions and store the details of the transaction for future queries. After receiving requests from clients, DTRM generates a list of prospected providers. The model only suggests the best possible providers based on their previous performance, though the final selection will be in the hand of the client. Both the provider and the clients have the option to inform about their experiences to the transaction register. Figure 3.2 shows the work-flow of the approach.

Though the DTRM can be implemented in any of the registries, it will be better to implement it in a Transaction Registry where possible. As transaction registry will keep all the transaction records, it will be effective if the model is implemented in it. On the other hand, DTRM not implementing in the service registry means it will reduce its work load. There are some advantages using a new registry, which are discussed bellow:

- The recommendation from the referrer can be biased and there is no good way to verify that. As the transaction registry stores all the records of previous transactions, we do not have to wait for the feedback from the recommender like in the distributed approach; rather we just need to request the transaction registry to know the previous performance of the provider. Using transaction registry this way will reduce the dependency on the recommendation.

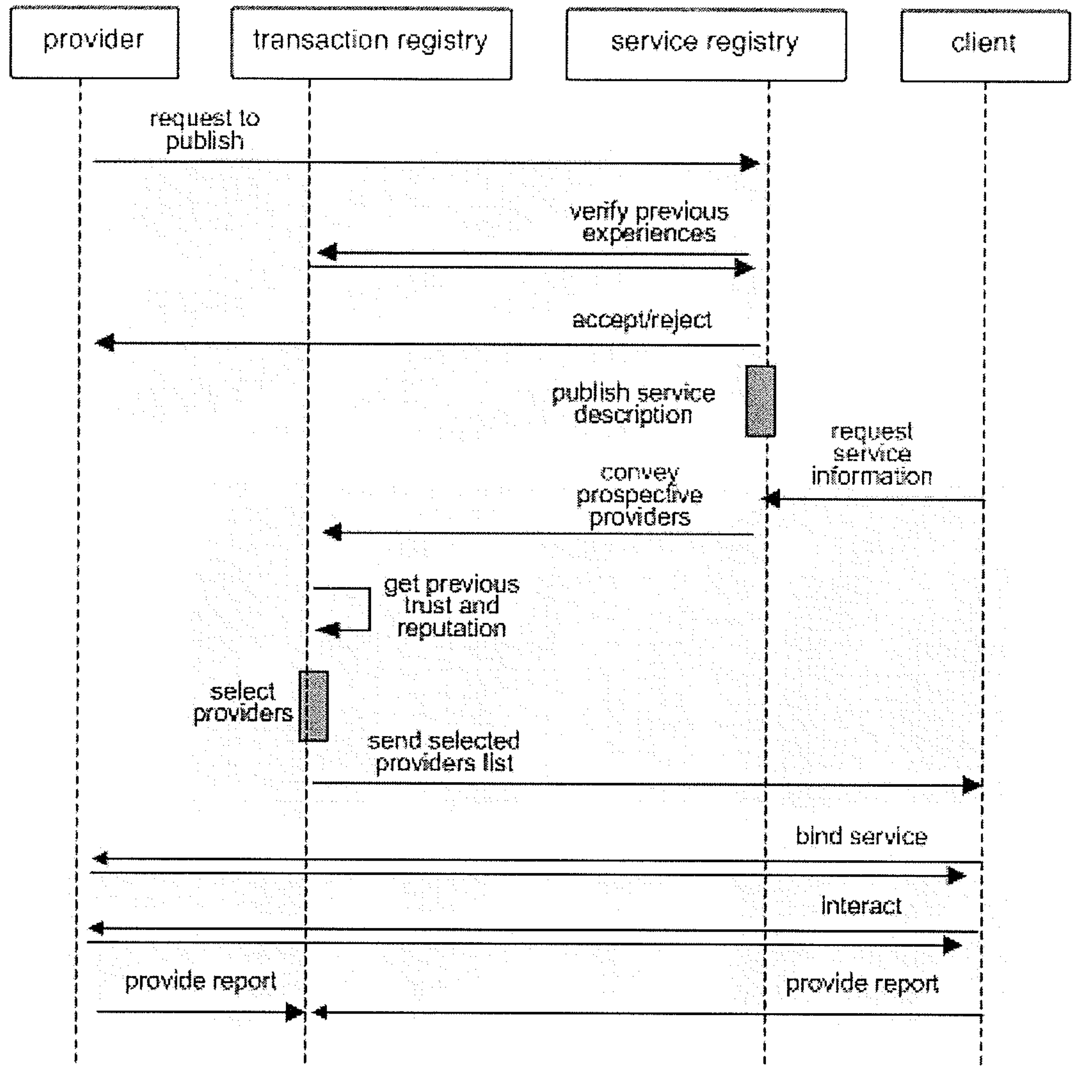

Figure 3.2: Centralized Approach

- The strength in centralized approach is that we need not to be dependent on the referrer's recommendation; rather the transaction registry can provide prospective providers considering their earlier transaction details.

- If we need to query on some provider or transaction, the transaction registry can help by providing the necessary information to the requester.

- The transaction registry can help checking the reputation of the referrer for authentication.

3.3 Distributed Approach

In the distributed approach, all the clients will be interconnected like a peer to peer architecture. Each client will keep track of its own experiences and share them upon request. Here, each client will act as a referrer for other transactions involving other clients. The Dynamic Trust and Reputation Management Model will be applied in every client's workstation. Figure 3.3 shows the work-flow of the approach. By using the concept of referrer one can use other's reputation about the provider to achieve better selection. To avoid biasedness of the referrer, 'social classification mechanism' or 'checking referrer's reputation mechanism' can be used.

The strength of the distributed approach is the classification of the clients of the society. If proper feedbacks of previous transactions are obtained then distributed approach can be an effective approach.

- As not all the referrers are equally credible the social classification process can be applied, which will minimize the fraud cases.

- The structure is the same as normal Web Services; therefore, working with it will be easy. Except the requester, all other clients can act as a referrer and share information.

- The distributed approach depends on the truthfulness of the referrer and thus on decision making. The referrer can provide biased feedback which can effect the provider selection. It will be hard to recognize reliable feedbacks in this way. We can only check the reputation of the referrer predicting that the referrers will share precise information which may not be the always true.

The distributed approach has some drawbacks as well.

- performance may suffer since not all referrers will reply instantaneously

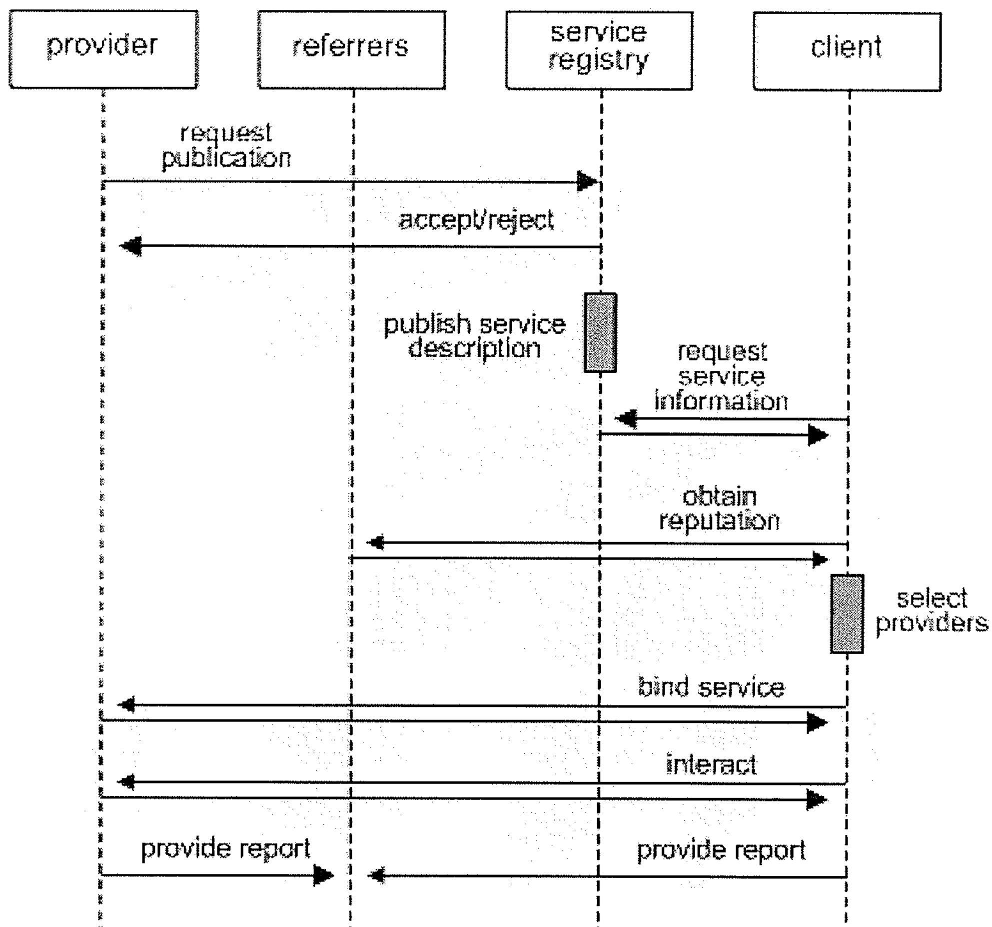

Figure 3.3: Distributed Approach

- the number of feedback reports received may vary (could be large or small)
- the distributed approach may be more difficult to secure against malicious intruders than the centralized one

3.4 Work-flow

The work-flow of the two approaches can be divided into three subcategories: a) how the service description will be published, b) how the selection process will be done and c) how the interaction and feedback will be handled.

3.4.1 Centralized Approach

- Publishing:

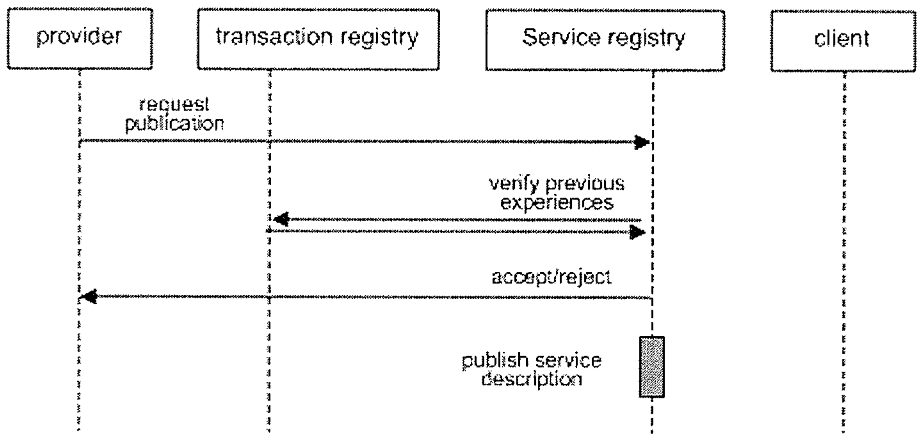

Figure 3.4: Centralized Approach: Publishing

A provider requests service registry for publishing the services it wants to provide. After that the service registry takes information from the transaction registry about the provider before accepting the request. Providers, who are having a acceptable satisfaction value, will get acceptance easily whereas others may have a conditional acceptance. This way, the providers with good reputation get the privilege in publishing. Later the provider publishes its service description. The publishing process is shown in figure 3.4.

- Selecting:

A client requests for the information of interested providers of a particular service to the service registry. The service registry sends the provider information to transaction registry and the client. For each of the providers the transaction registry gets the trust from the client and reputation from the previous transactions. After that the DTRM applied in the transaction registry and generates suggestion for the client. At last the transaction registry sends back the preference list of providers to the client.

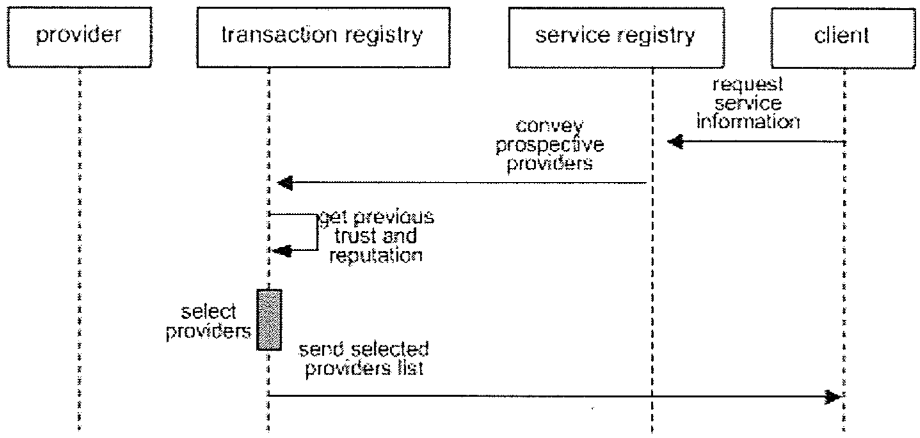

Figure 3.5: Centralized Approach: Selecting

In the Web Services, a service provider publishes the description of services it is going to provide. However, it can not be verified whether the provider has the ability to fulfill those claims or not. Yet the capability of the provider can be assumed by the previous performance of the provider. For this, it is essential to get authentic previous experiences about the service providers. The transaction registry is assigned the task of keeping track of previous information. It also stands behind to check whether the provider fulfills the promises as advertised to maintain good reputation based on the provider feedback. The selection process is shown in figure 3.5.

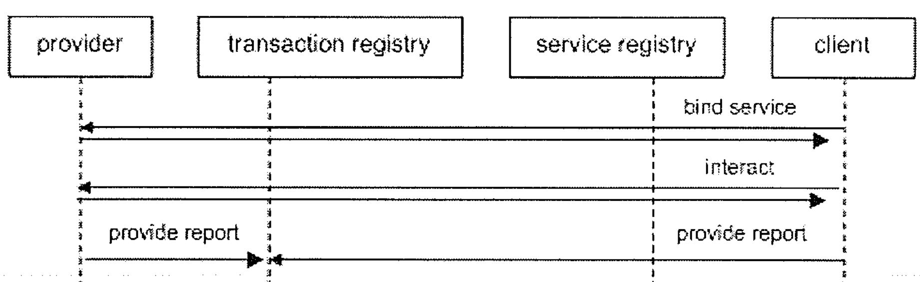

Figure 3.6: Centralized Approach: Reporting

- Reporting:

  After binding and interacting both the provider and the client may provide feedback to the transaction registry. These responses will help other peers while selecting providers in future. As satisfaction value depends on the provided service, the providers will try to increase its service to raise the satisfaction value and thus influence clients. The reporting process is shown in figure 3.6.

3.4.2 Distributed Approach

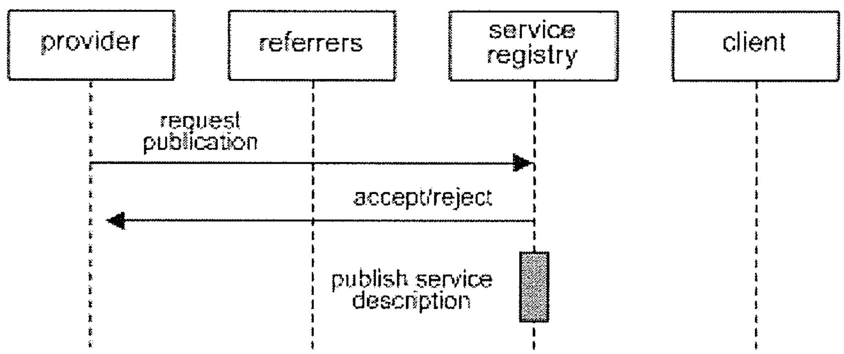

Figure 3.7: Distributed Approach: Publishing

- Publishing:

  A provider requests to publish the providing service description to the service registry. The service registry sends back reply to the request. If the provider does not maintain certain satisfaction value, then the request may be rejected or accepted with certain condition. In this manner, the reputed providers get the preference to provide support. Then the register will publish its service description. The publishing process is shown in figure 3.7.

- Selecting:

  After informed by the registry about the prospective providers, a client can communicate

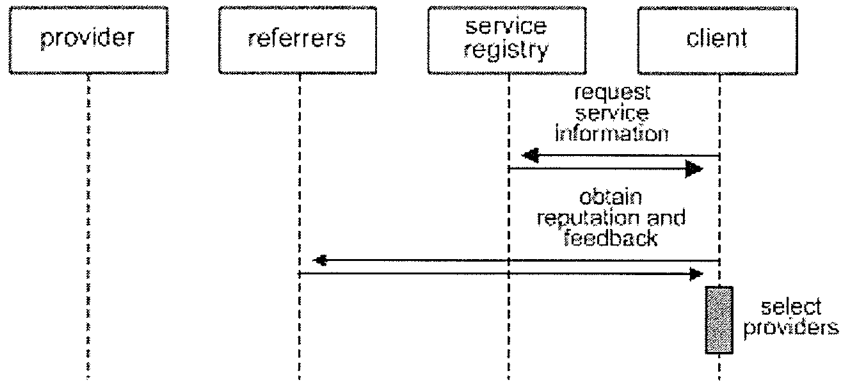

Figure 3.8: Distributed Approach: Selecting

with other peers about the service of those providers. As the process is distributed, clients have to depend on the feedback from other peers. Furthermore, not all feedback is equally credible; therefore some sort of classification must be used to simplify the situation. In order to facilitate this process the clients can be categorized into three different groups: family or relatives, friends and community. The DTRM assumes that relatives will provide more trustworthy feedback than friends, and friends will provide more trustworthy feedback than members of the general community. These differences are reflected in the model by providing separate weight for each of the components. Applying this, the occurrence of fraud cases can be minimized also.

The DTRM will be applied at the client end, from gathered feedback from other sources. The client classifies the feedback and generates a suggestion for the client. The client then follows the suggestion or chooses on its own from the list of providers. The selection process is shown in figure 3.8.

- Reporting:

Since in the distributed approach the referrer's feedback influence the client's selection process, providing feedback after every transaction is very important. Once a client selects

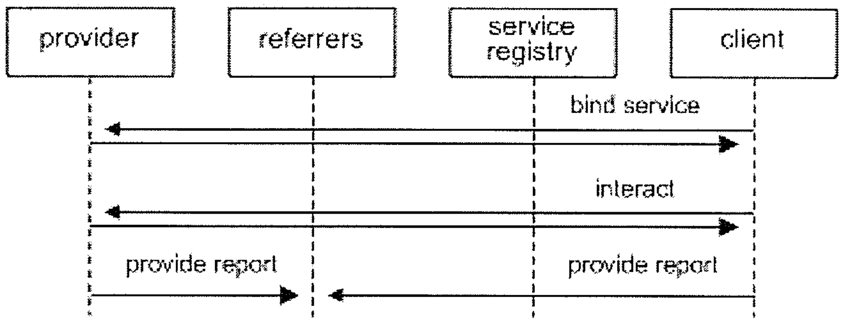

Figure 3.9: Distributed Approach: Reporting

a provider then the rest of the process is more straight forward. The client has to bind and interact with the selected provider. After binding and interacting, both the provider and the client will provide its own experience (optional for providers, mandatory for clients) to everyone. As trust and reputation value will depend on the service provided, the providers will try to increase their level of services gradually for a better satisfaction value. The reporting process is shown in figure 3.9.

Chapter 4

Results and Evaluation

We analyzed the performance of our Dynamic Trust and Reputation Management (DTRM). We have built a simulator to examine the behaviour of DTRM. A number of experiments are run with this simulator and the results have been analyzed to evaluate the DTRM.

We need to store the information of the providers, services and the clients and their transaction history. Hence, we maintain a database containing data tables. In this thesis we are using *db.** database. *db.** supports both network and relational database model. The advantages of using a network model are, it eliminates data redundancy by relating the records directly, without requiring the duplicate field and index file and both the network access and the indexed access are independent methods. By combining these technologies the database design flexibility is maximized [1]. In the network database model, the relationships among records are well defined and maintained through sets. "*A record is a named collection of related fields, which are stored and accessed as a unit*" [1]. Whereas "*a set defines a one-to-many relationship between two record types (member to owner)*" [1].

The database operation flow is given in the Figure 4.1. According to the figure the database is designed and the DDL specification is prepared before compiling the DDL specification. After that the code (which includes db.star.h) is designed and compiled. After compiling the code compiler we link the program with *db.** library to create the executable application program. Finally, we can run the *db.** application program [1].

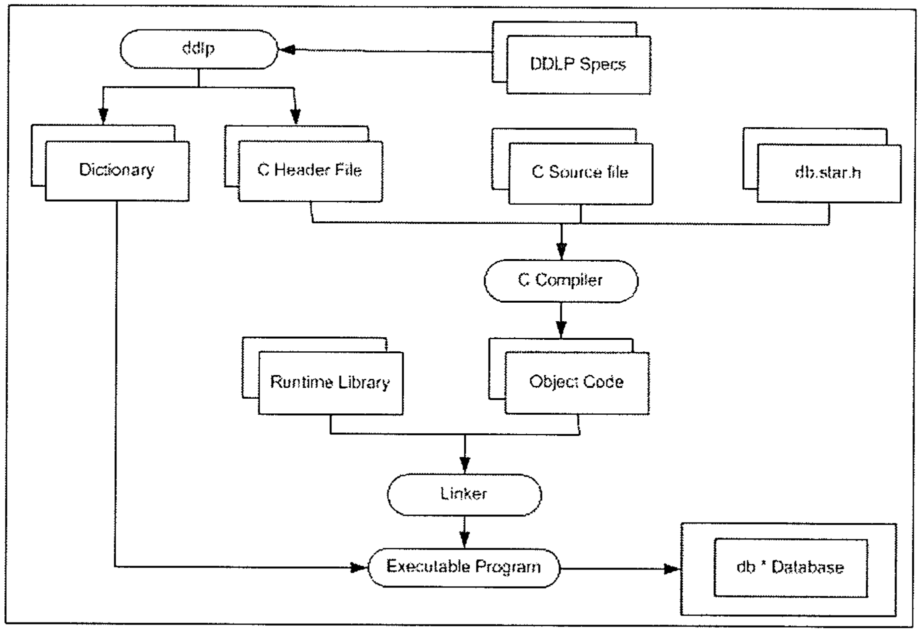

Figure 4.1: Database operation flow [1]

In this chapter, we are going to discuss the database design for this thesis. After that the initialization of simulation is discussed along with the social classification methods are described with a random example. After that several simulation processes and there results are given. The chapter ends with an evaluation on the experiments.

4.1 Database Design

Before running the simulator we need to initialize our database. The database contains several records and sets. The relation among the records and the sets is provided in Figure 4.2. In this thesis, we have designed our database with six records and six sets. Among the records three are owner records and rest of them are member records. The database needs to create the tables before the actual simulation runs.

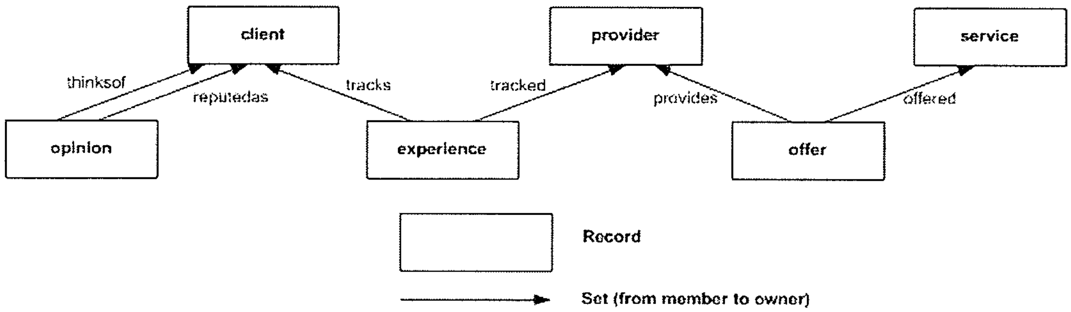

Figure 4.2: Database schema

4.2 Initialization

Certain initialization will be needed for the simulation run.

4.2.1 Setting up the participants

Our simulator will have N_c clients who are requesting services among the assigned N_s different services. The services are provided by N_p service providers, each of these providers is providing exactly P_s number of services. The providers will have a number of offers for the services. A provider can provide same service more than once with different offer.

We categorize the providers into three groups initially to match with the final outcome of the simulation results: most reliable group of providers, reliable group of provides and non–reliable group of providers. The trust and reputation value of each of the providers will be any value from 0 to 1. The groups are assigned by providing different range of weights for each of the transaction. The non-reliable providers can be a provider with low trust and reputation value, at the same time it can be a provider with fluctuated trust and reputation value. The groups are segregated by their credibility. We made the groups before running the actual simulation predicting that the group of most reliable providers will have most of the calls from the clients. After each of the experiments if the result matches as we predicted, it can be claimed that the DTRM works fine for the providers who maintain better trust and reputation.

4.2.2 Selecting options

The selection of the providers can be done randomly or non-randomly. Centralized and distributed are the two approaches which a client can choose for a non-random simulation. According to the model, the centralized approach uses the additional transaction registry to provide the necessary information for selecting the best provider. On the other hand, in the distributed approach we need to call for other people's opinion. As all the recommendations are not equally credible, we are going to classifying the society into groups. Therefore, we need a technique to classify the clients into groups. The groups can be created by using two different techniques: link based and area based.

4.2.3 Classifying clients for distributed approach

For distributed approach the social classification of clients is needed. Among the classified groups, the members of the family are the most trusted group members; then the friends and at last the members of the community, *i.e* the rest of the clients rather than the members of the family and friends. The participant clients are divided into different groups by either link based method or area based method.

- ***Link Based Method***

  In linked based method we create a connectivity graph by generating links between randomly selected node peers. The total number of links is determined by a predefined 'degree'. After that, we get the edges among the nodes and depending on that we classify the clients into groups. If the number of edges is less between a client and the requester, it can be assumed

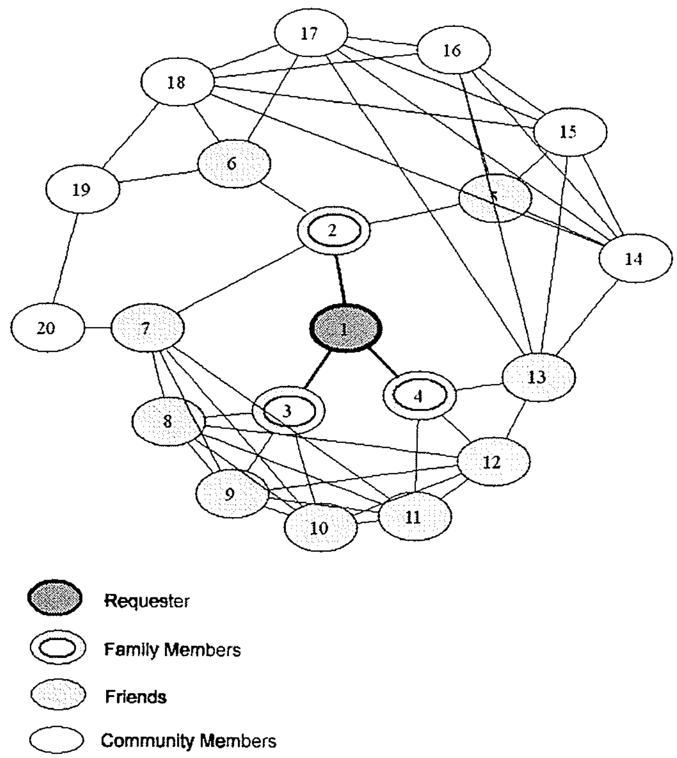

Figure 4.3: Linked based Social Classification (sample)

that the client have a close relation with the requester. These nodes will be assigned as the member of the family. Then according to the number of edges the friends and the community members are assigned.

For an example, for 20 nodes we select link based method. We consider node {1} to be the requester client node for the first cycle. After the mapping, the nodes with a one hop distance from the requester are assigned as family members, if the distance is two hops then those are assigned as friend and the rest are assigned as community members. In the example 50 edges created. After classifying the clients we have found 3 family members, 9 friends and 7 community members. The link based graph of 150 clients is very complicated and difficult to understand. For a clear understanding a small link based graph (Figure 4.3) of 20 clients is drawn where the edges are shown clearly.

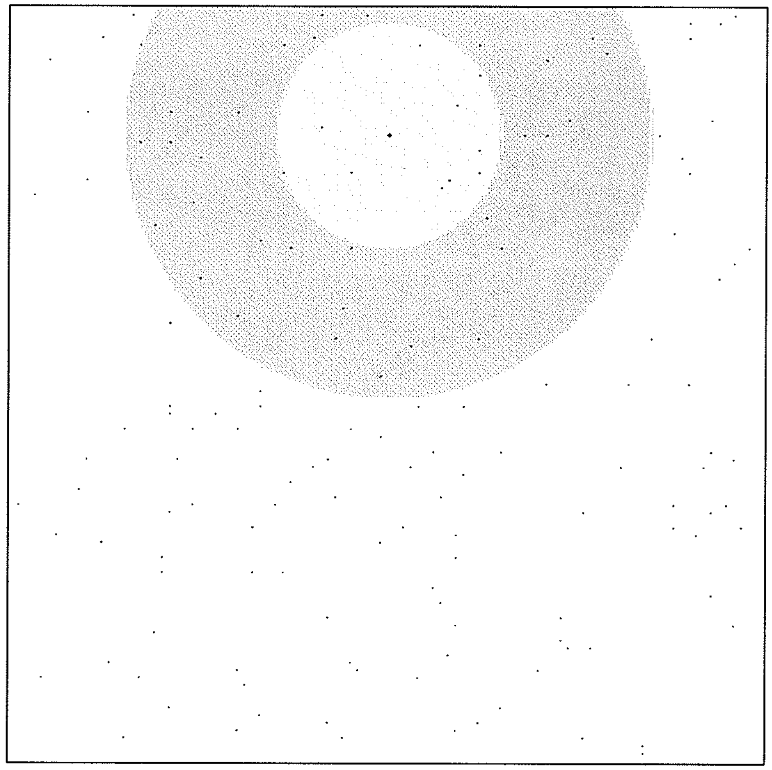

Figure 4.4: Area Based Social Classification (sample)

- *Area Based Method*

In area based method, we imagine a two dimensional graph as our scope and the clients are spread in the scope. Each of the nodes is denoted as a client. Initially a randomly selected node is assigned as the origin. This origin will act as a service requester. We map the distance of all other nodes from the origin and make groups according to their distances.

In each cycle of the simulations all the nodes will have their own circle of family, friends and community. For an instance of an example of 150 clients using area based method we find that, the node {50,83} is selected as the origin on a random basis for the first run. Then from that we map the distance of all other nodes and find 10 family members, 30 friends and 109 community members. The scope with the client nodes are shown in Figure 4.4.

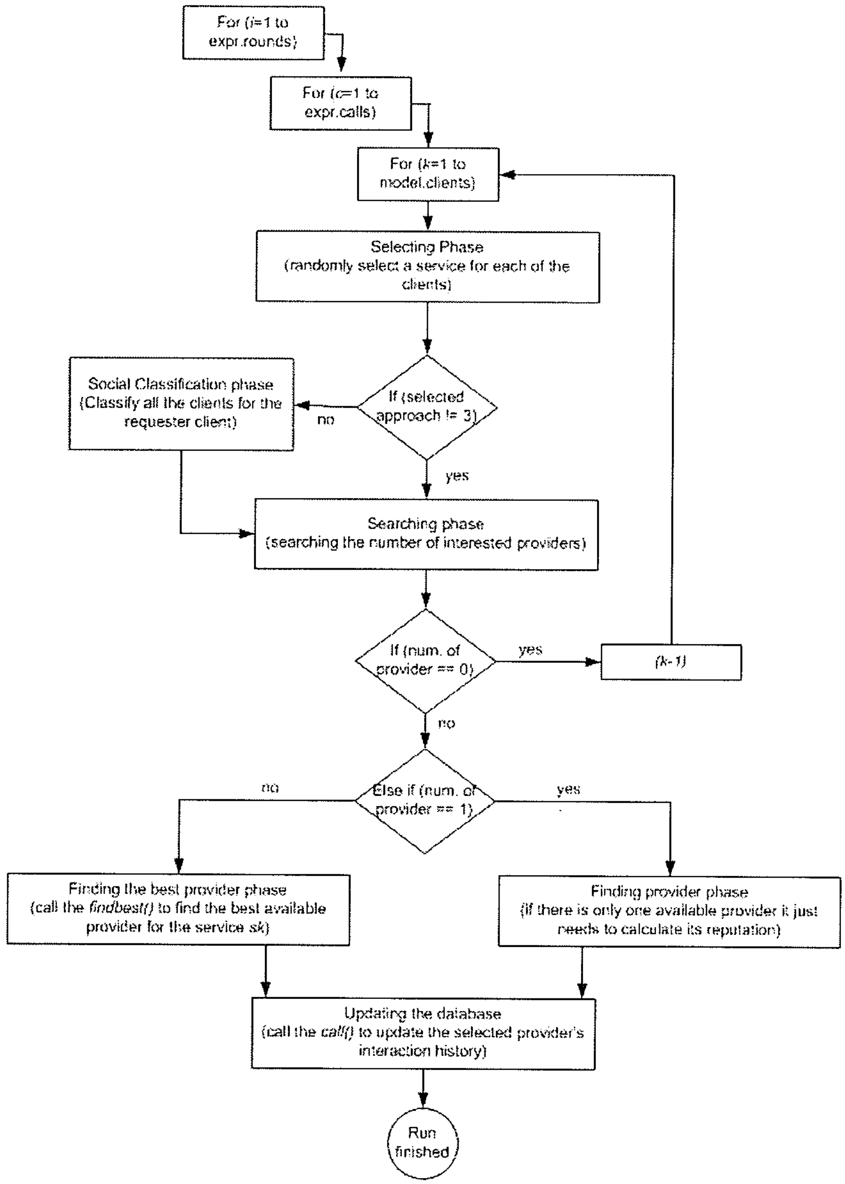

Figure 4.5: Progression of interactions

In the figure the nodes in the small circle are the members of the family and the nodes in the big circle are the friends of the origin node. The rest of the nodes are members of the community.

4.2.4 Setting up the simulation cycle

Several simulation runs are needed to obtain an ideal evaluation of our model. For our experiments we have run 20 rounds where each of the rounds has 200 calls. Therefore, the selection process is running 4000 times and producing results. We think that this many cycles are reasonable enough to produce an acceptable result.

In every experiment the providers declare the services they will provide. All the clients will select one random service. The corresponding providers and their reputation will be taking care for evaluation. The evaluation process will work if there is more than one service provider is available, otherwise the interested provider will be selected without any additional delay. After the selection process the database will be updated with the new trust and reputation value of the provider. The progression mentioned above is shown in Figure 4.5.

4.3 Experiments

The results and analysis of some experiments using the simulator are discussed bellow:

4.3.1 Experiment 1

In experiment 1,

- the providers and services are selected randomly,
- no trust and reputation concept is implemented.

The clients will choose services at a random basis. After running the experiment we consider the proportion of the total calls to the calls directed to the most reliable provider and the proportion

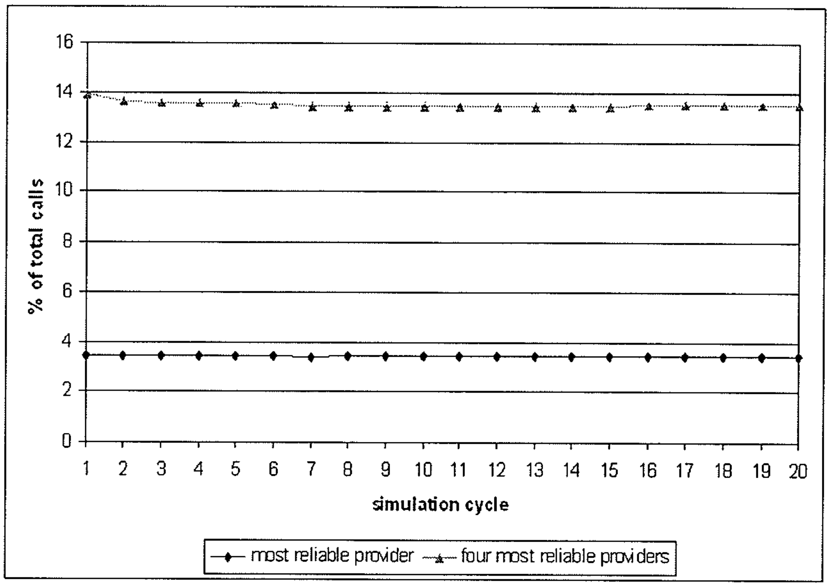

Figure 4.6: Percentage of random service calls to the most reliable provider and the four most reliable providers

of the total calls to the calls directed to the four most reliable providers. The graph is given in Figure 4.6, where it can be seen that the most reliable provider gets 3.43% and the four most reliable providers get 13.49% of the total service calls. The proportions are almost the same as a uniform distribution. Among the 30 providers the reliable providers did not get any advantages of their better trust and reputation values. Therefore, it can be predicted that all the providers, regardless of there credibility, get the same priority for service call.

4.3.2 Experiment 2

In experiment 2,

- services are selected randomly,
- providers are selected using centralized approach,
- only trust is implemented.

In this experiment we tried to find a suitable weight for the last few transactions depending on how

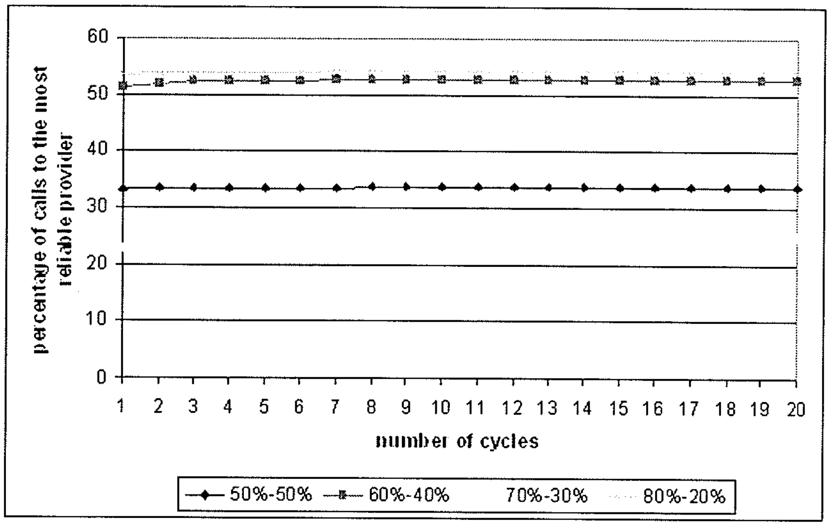

Figure 4.7: Comparison of percentages of calls of the most reliable provider using different weights for the last few transaction

much more we want to emphasize the recent transactions than the rest of the transactions. If 50% weight is given to the last k transactions then the *(t-k)* transactions will get 50% weight, which implies that we are considering all the transactions equally. If more than 50% weight is given to the recent transactions as we proposed, the rest of the transactions get less importance. However, we compared different proportions of weight (recent transactions-rest of the transactions): 60%-40%, 70%-30% and 80%-20% to choose the appropriate weight. According to Figure 4.7 when the weight is 50%-50% then the curve is almost straight and when the weight is 60%-40% and 80%-20% a minor improvement can be noticed. However, when 70%-30% weight is given the curve gets better than other curves. Therefore, we decide to choose 70%-30% weight for the last k transaction for rest of the experiments.

4.3.3 Experiment 3

In experiment 3,

- the services are selected randomly,

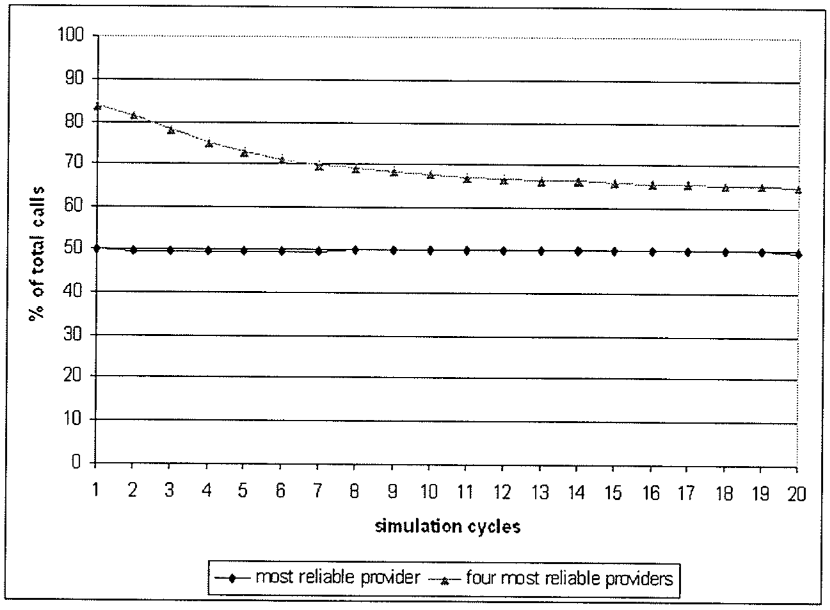

Figure 4.8: Percentage of calls using trust in centralized approach for the most reliable provider and the four most reliable providers

- providers are selected using centralized approach,

- only trust is implemented in this experiment.

In this experiment the centralized approach is used. Here the additional transaction registry is assumed to use which will have the track of all the previous transactions of the providers with each of the clients. After running the experiment we can see in Figure 4.8, that the most reliable provider gets almost 50% of the total service calls and the four most reliable providers started getting 83% of the service calls ended up with 65% of the service calls. As the services are chosen randomly and not all the services are offered by all the providers the four most reliable providers' curve came like that. The most reliable provider has a good transaction history and has a steady percentage of service calls.

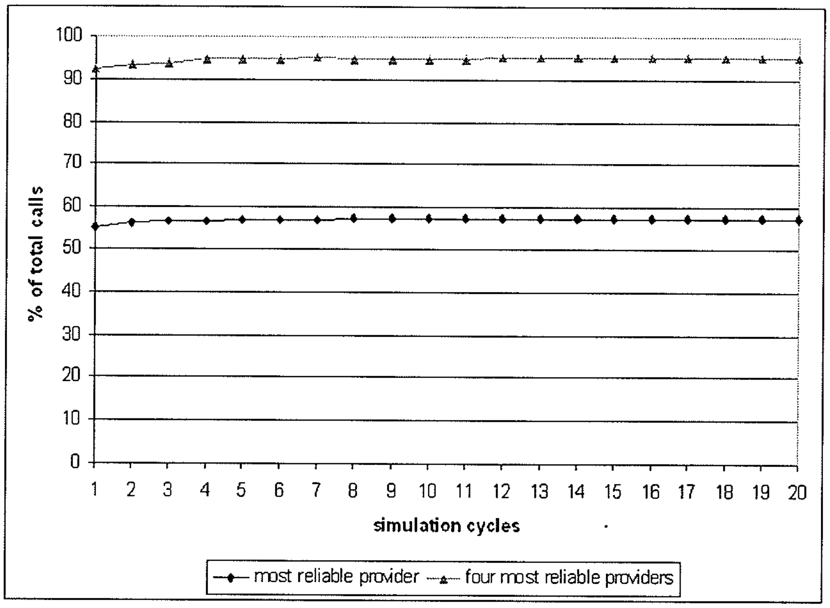

Figure 4.9: Percentage of calls using trust and reputation in centralized approach for the most reliable provider and the four most reliable providers

4.3.4 Experiment 4

In experiment 4,

- the services are selected randomly,
- the providers are selected using centralized approach,
- both trust and reputation are applied.

Trust and reputation are used in this experiment using centralized approach. We assume that every client had previous transaction records with all the providers. Therefore, each client has his own trust with all the providers. In centralized approach, the reputation of one provider is nothing but the simple arithmetical average of the other entire client's trust with that provider. According to Figure 4.8 we can see that the most reliable provider gets more than 55% of the total service calls and the four most reliable providers get 95% of the total service calls. From this Figure 4.8 it is clear that higher trust and reputation value can ensure more service calls. The four most reliable providers are called most of the times among which the most reliable provider is called more than half time.

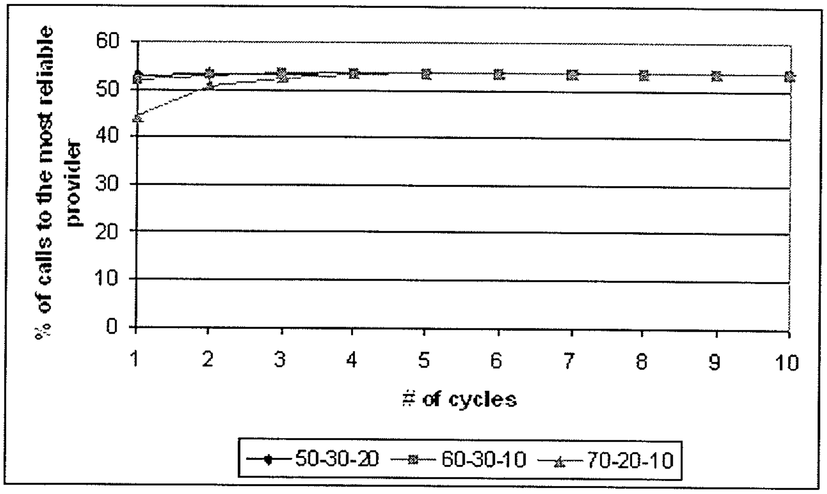

Figure 4.10: Percentage of calls to the most reliable provider using different weights for family, friends and community members in distributed approach

4.3.5 Experiment 5

In experiment 5,

- services are selected randomly,

- the providers are selected using distributed approach,

In this experiment we tried to decide the different weights for the family, friends and community members. According to our model we want to give more emphasis on the feedback from family members, then feedback from friends and then the feedback from community members. In Figure 4.10, we have taken three different values for the three social classified groups and run the experiment with the same settings. We can see that after the 5th cycle whatever value we take the most reliable provider gets almost same percentage of calls. Hence, we assume that any one of the types will serve our model properly. In the figure, when we take 50% weight for the family, 30% weight for the friends and 20% weight for the community then the curve proceeds with a little change while, when we take 70% weight for the family, 20% weight for the friends and 10%

weight for the community then the curve starts with a lower value and jumps up within a few cycles and becomes steady after that. Another curve is neither too straight nor with a sudden jump. Therefore, we are considering 60% weight for the family, 30% weight for the friends and 10% weight for the community for the rest of the experiments.

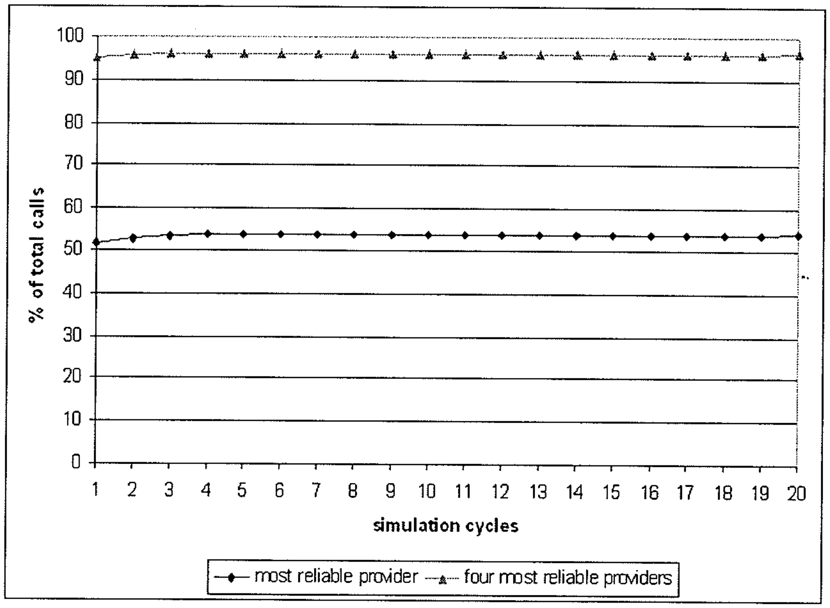

Figure 4.11: Percentage of calls using trust in distributed approach for the most reliable provider and the four most reliable providers

4.3.6 Experiment 6

In experiment 6,

- services are selected randomly,
- the providers are selected using distributed approach,
- only trust is implemented but no reputation is considered.

The distributed approach is applied in this experiment with the trust value of the provider for selecting the best service provider. In Figure 4.11, it is shown that the most reliable provider gets more than half, in particular near 53% of the total service calls. The four most reliable providers get around 96% of the total calls which means the rest 26 providers get only 4% service calls. In general, the requester clients look for reliable providers in other words the providers with a higher trust and reputation value. Therefore, it is obvious that they will call the reliable providers repeatedly.

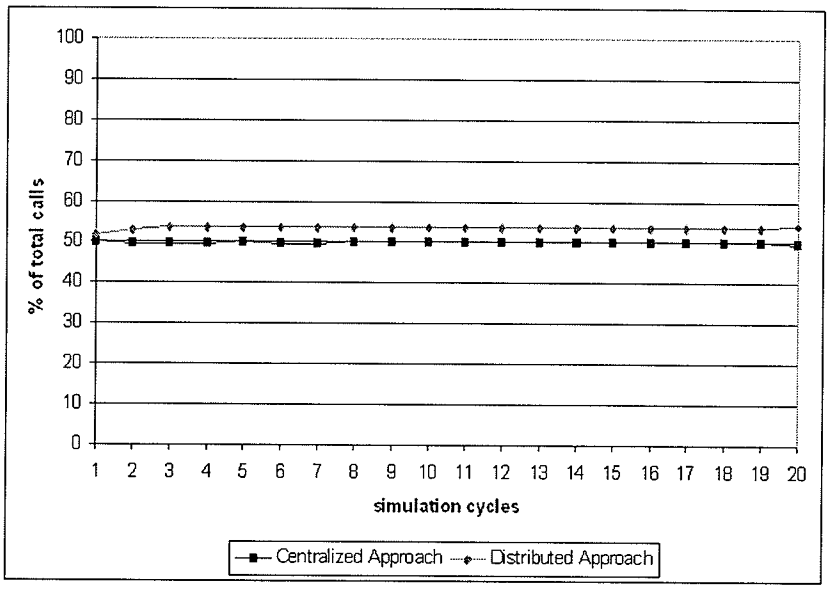

Figure 4.12: Comparison of calls of the most reliable provider using trust between centralized and distributed approach

4.3.7 Experiment 7

In experiment 7,

- services are selected randomly,

- only trust is implemented,

- no reputation is considered,

- only the percentages of total service calls to the most reliable provider are considered.

A comparison between centralized and distributed approach is made in this experiment. Centralized approach considers all previous transactions of the providers and distributed approach select providers according to the feedback of the classified social groups. In this experiment we made a comparison between the percentages of total service calls of the most reliable provider of both the approaches using trust. In Figure 4.12. the comparison is done. We ran the simulator twice to get the results to do this comparison. The experiments had their own set of data; therefore, there are minor differences between the curves (50% and 54%).

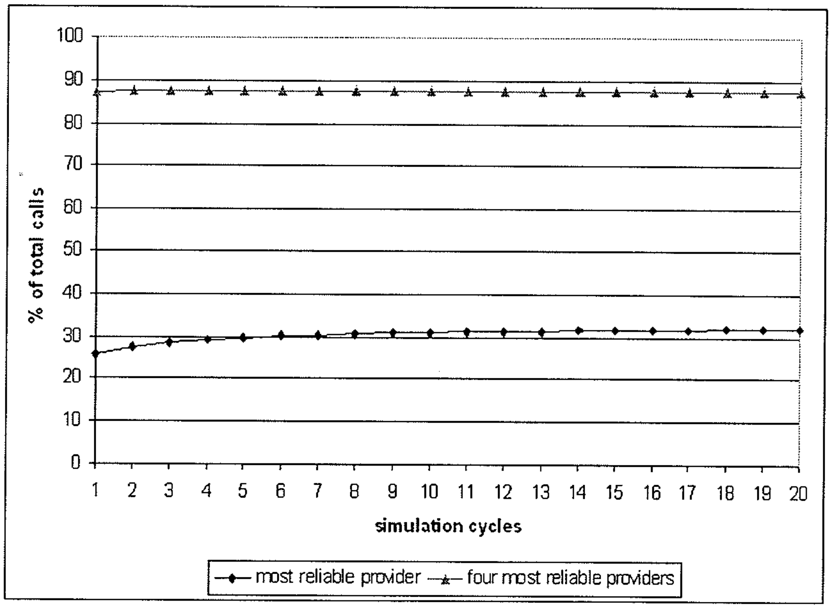

Figure 4.13: Percentage of calls using trust and reputation in distributed approach for the most reliable provider and the four most reliable providers

4.3.8 Experiment 8

In experiment 8,

- services are selected randomly,
- providers are selected using distributed approach,
- both trust and reputation concept is applied.

Distributed approach depends on the feedback of the clients. According to the social classification highest weight is given to the feedback from the family, then the feedback of the friends and at last the feedback of the community. Though in a real world scenario not all the clients from family, friends and community may not provide or interested to provide their feedback to the requester client. In this experiment we have considered all the feedback of the provider for selection. In this experiment the distributed approach has been applied using both trust and reputation. In Figure 4.13, it can be seen that at the beginning the most reliable provider got 25% calls and increased gradually. It also had an effect on the percentage of calls of the four most reliable providers.

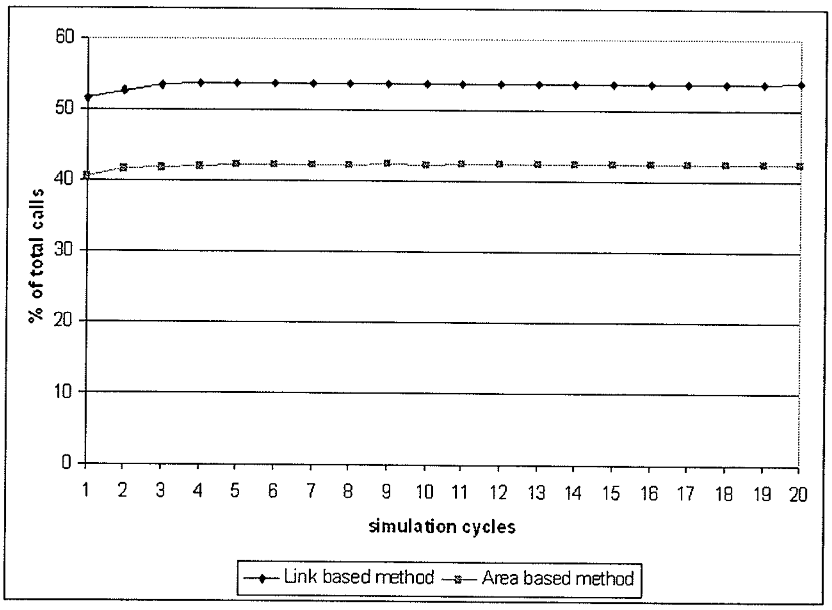

Figure 4.14: Comparison of calls of the most reliable providers of link based method and area based method in distributed approach

4.3.9 Experiment 9

In experiment 9,

- the services are selected randomly,
- only trust is implemented,
- only the percentages of total service calls to the most reliable provider are considered.

We draw a comparison between link based and area based method in distributed approach. Area based and link based approach are two ways of classifying the society into groups. In Figure 4.14 the comparison is shown. In the Figure the two carves are drawn using two distinct set of clients. Therefore the generated outcomes are different. However, the noticeable thing is that the progresses of the curves are almost same.

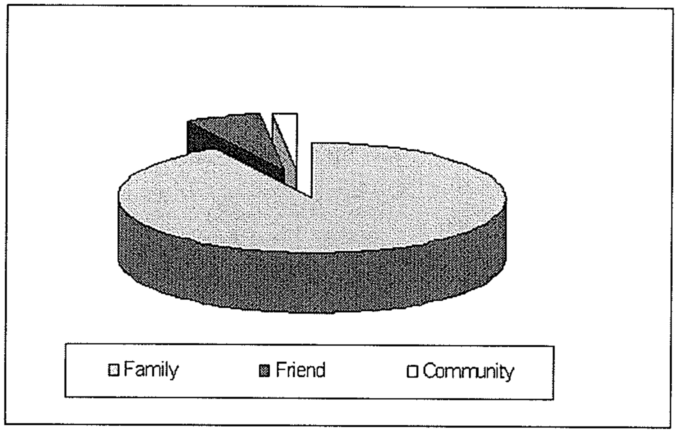

Figure 4.15: The proportion of calls among the provider groups

4.3.10 Experiment 10

In experiment 10,

- services are selected randomly,
- a comparison of the percentage of calls among the provider groups is made.

The providers are classified into three different groups according to their previous transaction history. Using the DTRM we make sure that the requester will get the service of the best provider. We can see in the Figure 4.15 almost all the calls (92%) are gone to the most reliable provider group. Then the reliable provider group got over 6% and the non reliable provider group got around 2% of the total service calls. This implies that the providers with good transaction history will get repeated service calls.

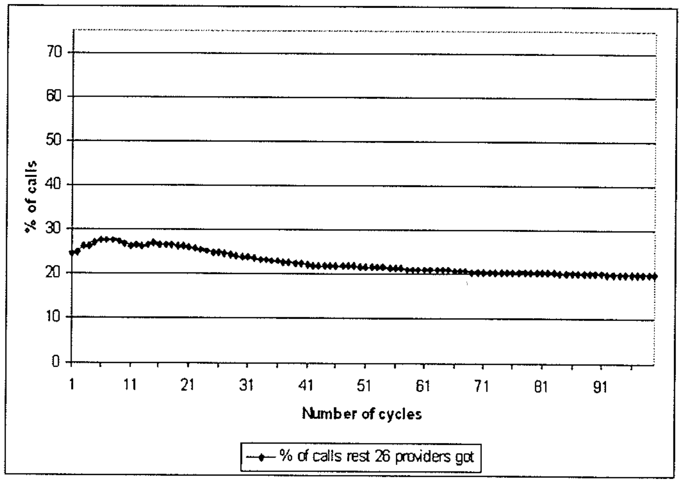

Figure 4.16: The percentage of calls getting by the reliable and non-reliable prover groups

4.3.11 Experiment 11

In experiment 11,

- services are selected randomly,

- providers are selected using centralized approach,

- only trust is implemented In this experiment the centralized approach is used.

We have considered a total of 30 providers and among them only 4 providers are in the most reliable provider's group. Rest of the providers are either reliable or non-reliable providers. In Figure 4.16 we have plotted the percentage of calls the reliable and non-reliable providers are getting. From the figure we can see that though on an average the 26 providers should get 87% of the calls, it is only around 25%. Furthermore, after 15th cycle it stated decreasing. This implies that by time the most reliable group of providers get more calls at the same time the rest of the providers get less calls.

Chapter 5

Conclusions and Future Exploration

In this thesis, a Dynamic Trust and Reputation Management (DTRM) model is proposed, which can recommend better service providers from a number of prospective providers using the trust and reputation concept. The selection process of existing methods considered issues such as credibility of the rates of the providers, while describing their models. In addition to those, DTRM uses effective classification of clients and an additional registry which offers an authentic transaction history. In the model the latest transactions are weighted more than the previous transactions.

5.1 Conclusion

Our proposed DTRM model emphasizes on the previous transactions of a particular provider while a client requests for a service. We conjecture that DTRM will help determining the satisfaction value which depends on the provider's trust and reputation. The most reliable provider will be selected for that service by using that satisfaction value.

The model is applied through a simulator we built and used to generate some results to analyze. At first we study an experiment in which we do not apply any approaches and methods. Then we carry out simulation studies in two different ways using two different approaches. Centralized and distributed are the two approaches we consider for DTRM. Using both of these approaches we implement only trust and trust and reputation.

In the studies we find that the most reliable providers group gets most of the calls as we stated in our hypothesis. Moreover, the most reliable provider gets most of the calls among all the providers. We carried out the experiments in both the centralized approach and distributed approach and found similar outcome. In the two methods of distributed approach: link based and area based, the percentages of selecting the most reliable provider are almost same. In another study we compare the interactions among the most reliable provider and the social classified clients and find that the interactions between the most reliable provider and the family members are more. This result implies that half of the family members recommended the most reliable provider.

The distributed approach takes more time than centralized approach for selection process as the approach involves classifying society into groups. Moreover, their satisfaction value is calculated according to the group it belongs to. Both the link based and area based approaches take almost same time to run in a similar setting.

Therefore, no matter which approach we choose or which method we select, the provider with good previous interaction history and with good recent transactions will be the most reliable provider.

5.2 Future Exploration

Future work includes a couple of issues those are extension of the DTRM model proposed in this thesis. Those are:

- We assume that all the clients will provide feedback to the requester. However, in real life this may not be the case. This issue can be considered for the future. Some questions should be clarified such as: what will happen if there is no feedback, how long a requester should wait before start the selection process, how the value of *'w'* will change over time depending on the received feedbacks.

- We can study the existing Web Services structure particularly WSDL and UDDI, and existing protocols and try to find out how the selection process and service discovery can be improved by using the social classification concept.

- We assume that a requester prefers accuracy rather that time meaning he does not have any problem to wait for a little long to get the best provider rather than get stuck with any provider. However, some requesters may prefer quick service rather than perfection. In our thesis we consider the best selection on time efficiency.

- In the DTRM model we mention how a client can improve his standing to a requester. For the improvement we only consider the last transaction or the most recent transaction of that client with the requester. Other methods can be used such as depending on the satisfaction value.

The DTRM will be helpful in overcoming some of the current limitations of Web Services and can turn Web Services into a trusted marketing technology over the net.

Appendix A

Appendix

A.1 DB.* ddl file

In *DB.** database management system 'Database Definition Language' (DDL) file will create a header file for the code. The ddl file of out simulator is given bellow.

```
database ppt {
    data file "ppt.d01" contains provider;
    data file "ppt.d02" contains service;
    data file "ppt.d03" contains offer;
    data file "ppt.d04" contains client;
    data file "ppt.d05" contains experience;
    data file "ppt.d06" contains opinion;
    key file  "ppt.k01" contains ckey, rkey;
    key file  "ppt.k02" contains xkey;
    key file  "ppt.k03" contains pkey, skey;
    key file  "ppt.k04" contains fkey;

    record provider {
        unique key long pkey;
        double          consistency;
        double          honesty;
        double          rating;
    }
    record service {
    unique key long skey;
    }
    record offer {
```

```
        long      sid;
        long      pid;
        double    promise;
        compound unique key fkey {
            sid  asc;
            pid  asc;
        }
    }
    record client {
        unique key int ckey;
        long           group;
        double         cheating;
    }
    record experience {
        long    kid;
        long    vid;
        long    tcount;
        double  tsum;
        double  tmean;
        double  rsum;
        double  rmean;
        compound unique key xkey {
            kid  asc;
            vid  asc;
        }
    }
    record opinion {
        long     mid;
        long     rid;
        long     distance;
        double   credibility;
        compound unique key rkey {
            mid  asc;
            rid  asc;
        }
    }

    set provides {
        order  asc;
        owner  provider;
        member offer by sid;
    }
    set offered {
        order  asc;
        owner  service;
```

```
        member offer by pid;
    }
    set tracks {
        order  last;
        owner  client;
        member experience;
    }
    set tracked {
        order last;
        owner provider;
        member experience;
    }
    set thinksof {
        order  asc;
        owner  client;
        member opinion by distance;
    }
    set reputedas {
        order  last;
        owner  client;
        member opinion;
    }
}
```

Bibliography

[1] Db.* overview. http://www.ittia.com/dbstar/dbstar.html.

[2] Merchant rating. http://www.pricegrabber.com/rating_getreview.php/r=2270 /m=4642352.

[3] Understanding feedback scores. http://pages.ebay.com/help/feedback/feedback-scores.html.

[4] Web services activity statement. http://www.w3.org/2002/ws/Activity, 2002-2006.

[5] UDDI executive overview: Enabling service-oriented architecture. www.oasis-open.org, October 2004.

[6] Web services architecture. http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/, February 2004.

[7] Farag A. Azzedin. *Trust modeling and its applications for peer-to-peer based systems.* PhD thesis, University of Manitoba, June 2004.

[8] Gary Bolton, Elena Katok, and Axel Ockenfels. Bridging the trust gap in electronic markets. Technical Report 2002-26, Max Planck Institute for Research into Economic Systems, Strategic Interaction Group, July 2002. available at http://ideas.repec.org/p/esi/discus/2002-26.html.

[9] Silvia Breban and Julita Vassileva. Using inter-agent trust relationships for efficient coalition

formation. In Robin Cohen and Bruce Spencer, editors, *Proceedings of the 13th Canadian Conference on AI*, May 2002.

[10] Ronald. S. Burt. Bandwidth and echo: Trust, information, and gossip in social networks. *Networks and markets*, pages 30–74, 2001.

[11] Sandeep Chatterjee and James Webber. *Developing Enterprise Web Services: An Architect's Guide*. Prentice Hall, November 2003.

[12] eBay. Member (diverslaptopparts) profile. http://feedback.ebay.ca/ws /eBayIS-API.dll?ViewFeedbackmyWorld=trueuserid=diverslaptopparts.

[13] Christopher Ferris and Joel Farrell. What are web services? *Commun. ACM*, 46(6):31, 2003.

[14] Karl D. Gottschalk, Stephen Graham, Heather Kreger, and James Snell. Introduction to web services architecture. *IBM Systems Journal*, 41(2):170–177, 2002.

[15] Heather Kreger. Web services conceptual architecture. *IBM Systems Journal*, May 2001.

[16] K. Kurbel and I. Loutchko. A framework for multi-agent electronic marketplaces: Analysis and classification of existing systems. In *International ICSC Congress on Information Science Innovations (ISI'2001)*, March 17-21 2001.

[17] E. Michael Maximilien. Self-adjusting trust and selection for web services. In *ICAC '05: Proceedings of the Second International Conference on Automatic Computing*, pages 385–386, Washington, DC, USA, 2005.

[18] E. Michael Maximilien and Munindar P. Singh. Reputation and endorsement for web services. *ACM SIGecom Exch.*, 3(1):24–31, 2002.

[19] Daniel A. Menascé. QoS issues in web services. *IEEE Internet Computing*, 6(6):72–75, 2002.

[20] V. B. Mišić. Service selection should be trust- and reputation-based. In *International Conference on Systems, Computing Sciences and Software Engineering (SCS05)*, pages 359–364, Bridgeport, CT, December 2005.

[21] Pierre Omidyar. Founders letter. http://pages.ebay.com/services/forum/feedback-foundersnote.html, 1996.

[22] Michael P. Papazoglou and Jean-jacques Dubray. A survey of web service technologies. Informatica e telecomunicazioni, University of Trento, 2004.

[23] David C. Parkes. Directions for reputation. Power Point Presentation, June 2004.

[24] Shuping Ran. A model for web services discovery with QoS. *SIGecom Exch.*, 4(1):1–10, 2003.

[25] Frank Leymann Tony Storey Sanjiva Weerawarana, Francisco Curbera and Donald F. Ferguson. *Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More.* Prentice Hall PTR, March 2005.

[26] Richard Sutton and Andrew Barto. *Reinforcement Learning.* MIT Press, Cambridge, MA, 1998.

[27] Min Tian, A. Gramm, H. Ritter, and J. Schiller. Efficient selection and monitoring of QoS-aware web services with the ws-QoS framework. In *WI '04: Proceedings of the Web Intelligence, IEEE/WIC/ACM International Conference on (WI'04)*, pages 152–158, Washington, DC, USA, 2004.

[28] Kim Topley. *Java Web Services in a Nutshell.* O'Reilly Media, 2003.

[29] Thomas Tran and Robin Cohen. Improving user satisfaction in agent-based electronic marketplaces by reputation modelling and adjustable product quality. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 828–835, Washington, DC, USA, 2004.

[30] Aaron E. Walsh, editor. *Uddi, Soap, and Wsdl: The Web Services Specification Reference Book.* Prentice Hall Professional Technical Reference, 2002.

[31] Lawrence Wilkes. The cost and benefits of web services and service oriented architecture. http://roadmap.cbdiforum.com/reports/roi, 2003.

[32] Bo Xu and Surya B. Yadav. Effects of online reputation service in electronic markets: A trust-based empirical study. In *Ninth Conference on Information Systems (AMCIS)*, pages 404–409, August 4-6 2003.

[33] Bin Yu and Munindar P. Singh. A social mechanism of reputation management in electronic communities. In *CIA '00: Proceedings of the 4th International Workshop on Cooperative Information Agents IV, The Future of Information Agents in Cyberspace*, pages 154–165, London, UK, 2000. Springer-Verlag.

[34] Giorgos Zacharia and Pattie Maes. Trust management through reputation mechanisms. *Applied Artificial Intelligence*, 14(9):881–907, 2000.

[35] Liangzhao Zeng, Boualem Benatallah, Anne H.H. Ngu, Marlon Dumas, Jayant Kalagnanam, and Henry Chang. QoS-aware middleware for web services composition. *IEEE Trans. Softw. Eng.*, 30(5):311–327, 2004.