ED-BeCT : An Enhanced Dynamic Behavioral Cloud Trust Model To Evaluate The Trustworthiness of The Cloud Service Provider

by

Sara MOAZZEZI EFTEKHAR

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
PH.D.

MONTREAL, OCTOBER 07, 2020

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Witold Suryn, Thesis Supervisor
Department of Software Engineering & Information Technology at École de technologie supérieure

Mr. Michel Kadoch, President of the Board of Examiners
Department of Software Engineering & Information Technology at École de technologie supérieure

Mr. François Coallier, Member of the jury
Department of Software Engineering & Information Technology at École de technologie supérieure

Mme. Elli Georgiadou, Independent external member
School of Engineering and Information Sciences at Middlesex University London

THIS THESIS WAS PRENSENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND PUBLIC

SEPTEMBER 14, 2020

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENTS

Conducting this PhD has been a momentous experience in my life and it would not have been viable to do without the guidance and support received from many people.

I would like to express my sincere gratitude to my research supervisor Prof. Witold Suryn for his constant support, patience, incentive, and tremendous knowledge. Many thanks for accepting me as a Ph.D. student. I am very grateful for his appropriate advice, valuable comments and support provided during these years of my Ph.D. He taught me how to believe in myself in spite of all the challenges. I thank him for his excellent contribution to make my work productive. Thank you for allowing me to grow in confidence and teaching me how to be a competent researcher.

I am extremely thankful to the members of my Ph.D. committee, Prof. Michel Kadoch, Prof. François Coallier and Dr. Ellie Georgiadou for spending their time to read this thesis and giving me their precious comments. Special thanks are due to all the departmental staff in particular « SARA Ateliers » and Mr. Prasun Lala for their help during these years of my study both academically and officially.

Very especial thanks to my loving family who has supported me in each step of my life. My father, Abbas who is an exceptional man, generous and so kind and with his unconditional love supported me throughout my career. My mother, Fariba is an extraordinary woman who always encouraged me to make my dreams achievable. My mother-in-law, Nadereh who with her infinite kindness taught me to be a strong woman and my late father-in-law, Ali who was an unexampled man in my life.

Last but not least, I want to express my deep gratitude to my husband and love of my life, Mehran who has always given me a hand to deal with the difficulties and with his continuous and endless love, backing and comprehension during my Ph.D, helped me to complete this thesis. You always helped me whenever I needed and encouraged me at times I thought that

there is not any chance to continue. I hugely acknowledge his contribution and profoundly appreciate his faith in me. I dedicate this thesis to him as a proof of my love and appreciation.

ED-BeCT: Un modèle de confiance dynamique de comportement dynamique amélioré pour évaluer la fiabilité du fournisseur de services cloud

Sara MOAZZEZI EFTEKHAR

RÉSUMÉ

L'infonuagique est une technologie en évolution qui fournit des services informatiques à la demande pour fournir des ressources évolutives de manière dynamique et économique. Les entreprises qui fournissent ces types de services permettent aux utilisateurs de stocker leurs données et d'échanger des informations avec un espace virtuel à distance. Cependant, ce fait soumet cette technologie à divers risques et les menaces à la sécurité. Ainsi, dans les environnements cloud, l'évaluation de la fiabilité est un défi de taille qui nécessite beaucoup d'efforts et de temps pour renforcer la confiance des utilisateurs dans le choix d'un fournisseur de services cloud fiable. La confiance a été étudiée par de nombreux chercheurs utilisant plusieurs techniques et évaluer diverses caractéristiques de confiance. En fait, conceptuellement, la confiance a une notion vague et dans l'infonuagique, elle implique diverses caractéristiques de confiance telles que la sécurité, fiabilité, auditabilité, mutualisation, etc. Bien que des recherches approfondies aient été menées sur la confiance dans les environnements cloud, aucune étude n'existe qui évalue la confiance avec des mesures standard sous différents angles. Compte tenu de ces problèmes, notre objectif dans cette thèse est de proposer un modèle de confiance basé sur les standards de qualité du système et des logiciels existants, ainsi que de considérer le comportement des services cloud pour le comparer a l'accord de niveau de service (SLA) offert pour filtrer les fournisseurs de services cloud qualifies. Dans le cadre d'une proposition de modèle Amélioration de la confiance dans le cloud comportemental dynamique (ED-BeCT), nous étudions les normes de qualité du système et des logiciels applicables pour trouver les caractéristiques de confiance standard. Nous analysons la littérature pour dériver les caractéristiques de confiance généralement reconnues applicables à l'évaluation de la confiance dans le cloud computing. De plus, nous dérivons des mesures appropriées pour chaque caractéristique de confiance afin d'augmenter la précision du modèle proposé dans l'évaluation de la confiance. Concernant les différents aspects de l'infonuagique, nous généralisons ces mesures de confiance pour les trois types de modèles de services cloud (à savoir Software as a Service, Platform as a Service, Infrastructure as a Service). Pour considérer le niveau d'importance de chaque caractéristique de confiance, nous appliquons la méthode de cohérence complète (FUCOM) pour calculer les pondérations appropriées. Dans cette méthode, les poids sont calculés en fonction des conditions de transitivité mathématique et des relations égales entre les poids et les priorités comparatives des caractéristiques de confiance. La valeur finale de la fiducie est déterminée en appliquant les pondérations calculées dans les valeurs normalisées de la fiducie dans la méthode SAW (Simple Additive Weighting). Nos travaux introduisent ainsi un modèle de confiance dans le cloud pour pallier les carences des modèles antérieurs proposés par d'autres chercheurs. Tout au long de cette thèse, les détails et l'analyse présentés du modèle montrent que le modèle

proposé est précis et possède les capacités appropriées pour calculer la confiance dans l'infonuagique et évaluer les services cloud fournis.

Mots-clés: modèle de confiance dans le cloud, caractéristiques de confiance, évaluation de la confiance, mesures de confiance, fiabilité

ED-BECT: An enhanced dynamic behavioral cloud trust model to evaluate the trustworthiness of the cloud service provider

Sara MOAZZEZI EFTEKHAR

ABSTRACT

Cloud computing is an evolving technology providing the delivery of on-demand calculating services to offer scalable resources dynamically in an economical manner. The companies that provide these types of services enable users to store their data in, and exchange information with a virtual space remotely. However, this fact makes this technology prone to various risks and security threats. Thus, in cloud environments, assessing the trustworthiness is a momentous challenge that requires much effort and time to build up users' confidence to choose a trustworthy cloud service provider. Trust has been studied by many researchers using several techniques and evaluating various trust characteristics. In fact, conceptually, trust has a vague notion and in cloud computing, involves diverse trust characteristics such as security, reliability, auditability, multi-tenancy and so on. Although extensive research has been carried out on trust in cloud environments, no single study exists which evaluates trust with standard measures from different perspectives. In view of these issues, our objective in this thesis is to propose a comprehensive cloud trust model based on existing system and software quality standards as well as to consider the cloud services behavior to be compared with the proposed service level agreements (SLAs) to filter qualified cloud service providers. In the context of a proposal for an Enhanced Dynamic Behavioral Cloud Trust (ED-BeCT) model, we study the applicable system and software quality standards to find standard trust characteristics. We analyze the literature to derive the commonly recognized trust characteristics applicable for evaluating trust in cloud computing. Additionally, we derive proper measures for each trust characteristic to increase the accuracy of the proposed model in trust assessment. Concerning the different aspects of cloud computing, we generalize these trust measures for the three types of cloud service models (i.e. Software as a Service, Platform as a Service, Infrastructure as a Service). To consider the level of importance of each trust characteristic, we apply full consistency method (FUCOM) to calculate appropriate weights. In this method, the weights are calculated based on the mathematical transitivity conditions and equal relations between the weights and the comparative priorities of the trust characteristics. The final value of the trust is determined by applying the calculated weights in the normalized values of the trust measures in the SAW (Simple Additive Weighting) method. Our work thereby introduces a cloud trust model to overcome the deficiencies of the prior models proposed by other researchers. Throughout this thesis, the presented detail and analysis of the model shows that the proposed model is accurate and has the appropriate capabilities to calculate trust in cloud computing and evaluate the provided cloud services.

Keywords: cloud trust model, trust characteristics, trust evaluation, trust measures, trustworthiness

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABREVIATIONS

IaaS	Infrastructure as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
CSP	Cloud Service Provider
CSU	Cloud Service User
CC	Cloud Computing
CTM	Cloud Trust Model
SLA	Service Level Agreement
QiU	Quality in Use
QoS	Quality of Service
KPIs	Key performance Indicators
Ind	Individual
Org	Organization
AWS	Amazon Web Service
IT	Information Technology
CSA	Cloud Security Alliance
CAI	Consensus Assessments Initiative
AHP	Analytic Hierarchy Process
TM	Trust Management
IOWA	Induced Ordered Weighted Averaging Aggregation
SMI	Service Measurement Index

TMM	Trust Mining Model
ACO-BTM	Ant Colony Optimization algorithm Behavior Trust Model
PocT	Policy as a Trust Management Technique
RecT	Recommendation as a Trust Management Technique
RepT	Reputation as a Trust Management Technique
PrdT	Prediction as a Trust Management Technique
TTs	Trust Tickets
TPM	Trusted Platfom Module
TVEM	Trusted Virtual Environment Module
PLT	Prepositional Logic Terms
DTV	Direct Trust Value
PCC	Pearson Correlation Coefficient
ROI	Return On Investment
ITIL	Information Technology Infrastructure Library

INTRODUCTION

0.1 Research subject

The past decade has seen the rapid development of cloud computing in various industries. From the viewpoint of the National Institute of Standard and Technology (NIST) [2], cloud computing is an evolving technology for enabling ubiquitous, convenient and on-demand network access to shared configurable resources such as networks, servers, storage, applications and services that are capable to be rapidly provisioned and released with minimal effort. Among all attempts to define cloud computing such as the ones explained in the related ISO/IEC standards, the definition of cloud computing by NIST has been mostly accepted and the broad use of this definition in scientific publications (e.g. [3-6]) can be a good proof of its comprehensibility. According to the definition of cloud computing by NIST, it encompasses five essential characteristics (on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service), four deployment models (private cloud, community cloud, public cloud, hybrid cloud) and three service models (Software as a Service, Platform as a Service, Infrastructure as a Service).

Typically, cloud service models are presented in the definition of NIST [2] as follows:

- Software as a Service (SaaS) : Users apply the deployed applications by providers through either an interface such as a web browser or a program interface. Users are not able to manage or control cloud infrastructure including storage, servers and network;
- Platform as a Service (PaaS) : Users are able to deploy their own created applications or develop the applications using the programming languages, libraries and tools supported by providers onto the cloud infrastructure;
- Infrastructure as a Service (IaaS) : Users are enabled to provision the fundamental computing resources such as storage and networks to deploy the arbitrary software included operating systems and applications.

Considering the cloud computing nature and its several noteworthy features, more and more stakeholders, organizations or individuals are willing to outsource partially or completely the processes of their businesses to the cloud environments [7]. Trust in cloud infrastructure can play an important role in addressing the issue of cloud computing adoption and has received considerable critical attention from both academia and industry [8-10]. Although, to establish trust in cloud infrastructure different requirements of diverse users should be considered and it might be complicated [8], measuring trust and evaluating cloud services are the major challenges. The obstacles that can be manifested in security, availability, reliability, usability and referred to as quality of the proposed cloud services, have the main role in adopting cloud technology. The root of all these concerns is trustworthiness of the cloud service providers.

The dynamic nature and the fact that data which is the most valuable asset in the world of IT, is scattered in the cloud environments and will be processed and retrieved from various storage in remote locations [11], which is not under control of cloud customers, can compromise safety. In addition, regarding the continual growth of cloud service providers, selection of a provider that is able to meet all the users' requirements would be a decision-making problem. Thus, it is important to apply a precise method to solve this problem. Several cloud trust models are designed to eliminate these concerns and help both cloud service users and cloud service providers to have an effective collaboration. Cloud trust models by evaluating trustworthiness are able to help specify deficiencies in the cloud services. Consequently, cloud service providers can detect these deficiencies to proceed to improve their services and also cloud service users can select cloud services more accurately.

0.2 Trust and trustworthiness

It has commonly been assumed that trust is a sophisticated phenomenon [12-14]. Not only are there various definitions of trust in the literature, but also people have different attitudes toward it [13-15]. Basically, in cloud computing, defining the notion of trust depends on realizing functional and non-functional quality requirements. This shows a need to be explicit about exactly what is meant by the word 'trust'.

Broadly speaking, trust can be defined as a bilateral relationship between a cloud user and a cloud provider. Previous studies mostly defined ‘trust’ as a psychological state that consists of [12-17] :

- Expectancy- that is related to a particular behaviour from the trustee which is expected by a trustor (e.g. performing the required tasks effectively);
- Belief - the trustor is in the belief that based on the evidence such as competence and goodwill of the trustee, expected behaviour will occur;
- Taking the risk- based on this belief, the trustor is able to take risk.

It is a widely held view that trust is an uncontrollable and intangible concept [13, 14, 18]. There is a great volume of published studies describing the roles of many influential characteristics contributing to establishing a trust relationship [9, 13, 14]. What we know about trust in cloud computing is largely based upon empirical studies that investigate how these characteristics impact the relationship between two entities in cloud environments [14].

On the other hand, it is now well established from a variety of studies that trust is a measure of trustworthiness [13, 14, 19]. As Suryn in [19] argues, “Trustworthiness is an attribute of an entity deserving of trust or confidence, being dependable and reliable.” Regarding this definition, although Suryn in this work identifies credibility, reliability and dependability as the three significant characteristics for trustworthiness, convincing the users, particularly the corporate industrial customers, to broadly adopt cloud technology requires a larger list of characteristics [14, 19, 20].

A trustworthy cloud service provider will be determined based on assessing its trustworthiness and as mentioned earlier, trustworthiness can be measured by trust [14]. Therefore, a possible explanation for the existing relationship between trustworthiness and trust in cloud environments can be depicted as in figure 0-1 [14]. It is almost certain that the more evaluated characteristics, the greater trustworthiness for the cloud service provider can be achieved [14].

In other words, by measuring these characteristics that are the users' requirements representatives, potentially a trustworthy cloud service provider can be identified [14].

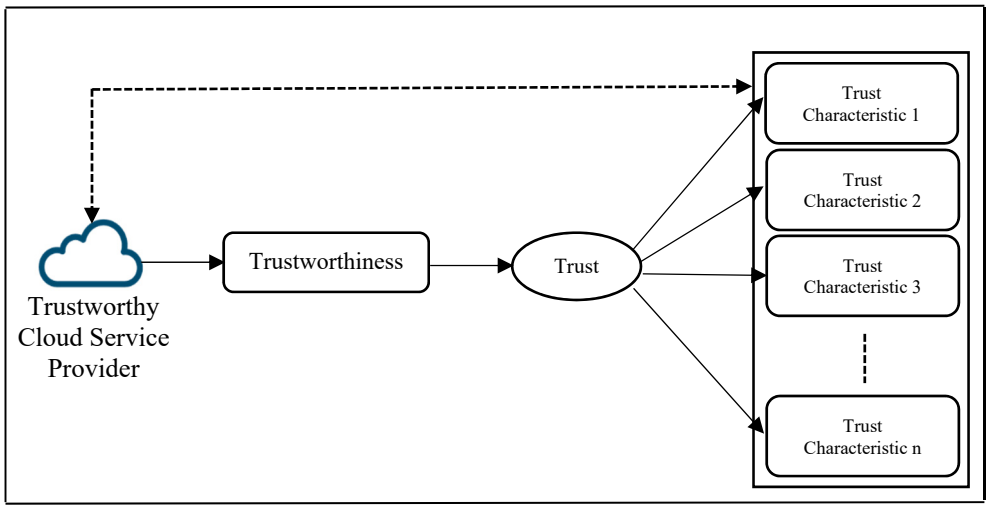

Figure 0-1. The relationship between trustworthiness, trust, and a trustworthy cloud service provider [14]

0.3 Problem statement and motivation

In the new global information technology, trust has become a central issue for many organizations and individuals [14]. Evidence suggests that trust is among the most important factors for establishing a relationship and adoption of technology such as cloud computing [14]. Although, more than two decades has elapsed from the first day of introducing cloud computing, unlike the various evolutions and progressions in this technology, trust and trustworthiness are still major concerns and open issues [14]. There is a growing body of the literature that recognizes the importance of trust and evaluating the trustworthiness of the cloud service providers (e.g. [21-23]) [14]. The main challenge faced by many researchers is that the notion of trust is combined with uncertain concepts and at the same time associated with functional and non-functional requirements [14]. This causes that trust in cloud computing remains an obstacle for cloud technology adoption [14].

On the other hand, there is no consensus based rule or agreed standard to evaluate trust in cloud environments [13, 14, 20]. In cloud computing standards, the key characteristics of cloud have been defined and several concepts related to this technology have been explained [14, 24]. However, there is no clear path to guide stakeholders or individuals toward a trustworthy cloud service provider [14]. To identify the influential characteristics that cover different aspects of trust and users' requirements, many researchers proposed several trust characteristics, but they are not sufficient enough to boost users' confidence in applying cloud technology [14]. For example, in the majority of scientific publications, security is first and foremost trust characteristic that needs to be tackled from different perspectives (e.g. [25]) [14]. Undoubtedly, security is a major challenge in cloud computing, but it is important to bear in mind that it is one of the various aspects of trust [14].

Moreover, a significant part of a reasonable evaluation of trust in cloud environments pertains to evaluating the measures of quality characteristics that are proposed as trust characteristics [14]. Most studies in the field of evaluating trust in cloud computing have only focused on the insufficient set of characteristics with inappropriate measures [14]. In addition, these measures are not necessarily based on the quality standards, which may lead to incomprehensive evaluation of cloud services [14]. Hence, extensive research has shown that there is an urgent need to address these trust characteristics along with their measures which can be supported scientifically [14].

Regarding the aforementioned criteria, a broadly acceptable cloud trust model is required to aid the assessment of cloud services that are proposed by cloud service providers. Despite the fact that several approaches have been identified to design a cloud trust model for assessing the trustworthiness of a cloud service provider, further research is needed to design a broadly acceptable cloud trust model that can evaluate various aspects of cloud services.

Therefore, the aim of this thesis is to design a broadly acceptable cloud trust model that to the best of our knowledge would be the first cloud trust model that can cope with different cloud

service users with different requirements and at the same time can assess standard trust characteristics. Figure 0-2 illustrates the limitations which are addressed in this research work.

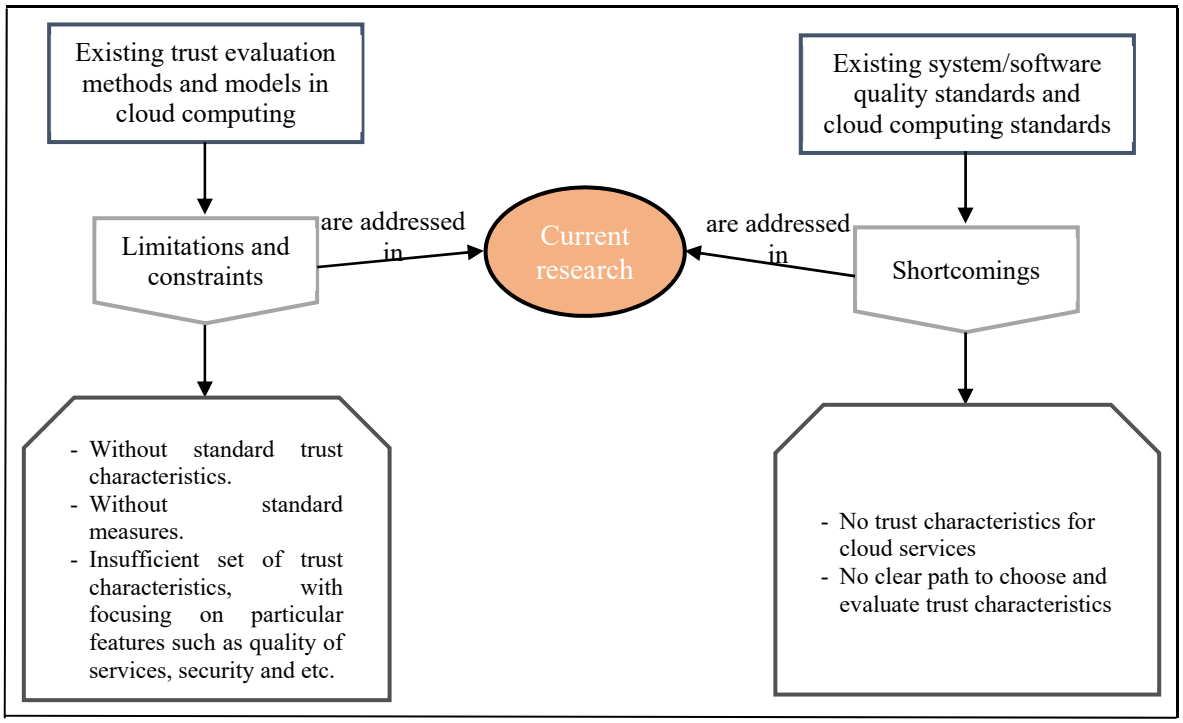

Figure 0-2. Research problems

The main motivations for this research project are :

- The need to map quality of service characteristics and the quality characteristics explained in system/software quality standards with the notion of trust and cloud computing;
- The need to support cloud service users in cloud computing adoption;
- The need to support cloud service providers to provide a trustworthy cloud service.

0.4 Significance and expected benefits of this research project

As already explained, this research project will overcome the limitations and shortcomings from which existing cloud trust models and related ISO/IEC standards suffer (figure 0-2).

In addition, this research project has some significant added values that are explained as follows:

- The proposed model will be beneficial for the cloud auditors and professional organizations in this field for the consideration in auditing cloud services;
- This research, once officially published will be used by ISO/IEC JTC1 SC7 WG06 as the supporting material for the ongoing work on the applicability of ISO/IEC 25000 SQuaRE series to Cloud Computing technology. Also, it will be used as the supporting material in the works of ISO/IEC JTC1 SC7 WG06 Study Group on Software Quality Engineering domain within SQuaRE chaired by Prof Witold Suryn, École de technologie superieure;
- The research solution will provide the various fields of industries and stakeholders a comprehensive cloud trust model that is able to support cloud service users in selection of a trustworthy cloud service provider, as well as cloud service providers to provide the trustworthy cloud services.

0.5 Reseach objectives

With respect to all the challenges that can arise by adoption of cloud computing, the ultimate goal of this research project is to develop a broadly acceptable cloud trust model that is applicable to evaluate trust from different perspectives (i.e. quality of services, conformity with users' requirements and trust characteristics) in cloud environments.

To make this goal attainable, we can classify the research objectives as follows:

- Identifying the key trust characteristics to be considered as complete as possible trust characteristics reference (in the time of conducting this research) that could be recognised in the system and software quality standards and cloud computing standards by analyzing:
  - the trust characteristics evaluated by existing cloud trust models in the literature;
  - system and software quality standards;
  - cloud computing standards.
- Analyzing and specifying the weaknesses of the current cloud trust models:

- Identifying the models that they have certain consensus published in renowned journals and software quality publications;
- Comparing them to see what they have in common and what are their empty spaces by matching with cloud computing standard (ISO/IEC 17788).

The methodology adopted in this research project to fulfill these objectives consists of five major phases: analysis, matching, tracing, development, exploitation. We define these phases briefly as follows.

The analysis phase is related to rigorous investigations and analysis in the literature. The particular objectives of this phase are to identify the trust characteristics and their measures that are proposed by various researchers, the weaknesses of existing classifications of trust models and the proposed methods for the trustworthiness assessment of cloud service providers.

In particular, in this phase we try to answer to the following research questions:

- What are the trust characteristics in cloud environments?
- What are their measures?
- Are the trust characteristics in cloud environments supported scientifically?
- Is there any classification for existing cloud trust models? If yes, what are they?
- What type of methods are proposed in the literature to evaluate a vague concept like trust in cloud computing?

The matching phase includes two stages; in the first stage, we match the characteristics that are identified in the previous phase with the characteristics that are explained in relevant ISO/IEC standards to extract all the prerequisite characteristics. Basically, the essential research question in this stage is: what are the main sub-characteristics for having an accurate evaluation?

In the second stage, we match the measures of the specified characteristics and sub-characteristics with the ones explained in ISO/IEC 25022 and ISO/IEC 25023, and propose certain measures for the characteristics without any defined measures in these standards by matching their definitions in ISO/IEC 25010, ISO/IEC 25011, ISO/IEC 17788, ISO/IEC 19086-1 with the proposed measures. In this stage, we try to find an answer for the question of what are the standard measures for the trust characteristics?

The tracing phase explains the differences and the similarities of IaaS, PaaS and SaaS and describes the feasibility of the cloud services assessments regarding the provided measures by different categories of users. Since one of the objectives of this research is to propose an applicable cloud trust model, it is substantial to consider the categories of users who wish to benefit from cloud services.

Accordingly, there are some research questions that need to be replied:

- How to categorize the different cloud service users?
- Are the features of three well-known cloud service models (Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)) influential in cloud service measurements?
- If yes, what are these influential features?
- Are all the proposed measures extracted from the system and software quality standards applicable and measurable in cloud computing?
- Which of the proposed measures is/are proper for IaaS, PaaS and SaaS?

The development phase explains the steps of the proposed model which is named as Enhanced Dynamic Behavioral Cloud Trust model (ED-BeCT). The main objective of this phase is to propose a capable cloud trust model according to the results of previous phases.

In exploitation phase, we demonstrate the phases of the ED-BeCT in an example. The basic objective of this phase is to achieve a better understanding of the proposed model and its core idea.

0.6 The structure of this thesis

The overall structure of this research project takes the form of 11 chapters, including:

Chapter 1 presents the literature review on fundamental concepts, main definitions and provides a critical appraisal of previous works in the domain of trust in cloud computing. In this chapter, the shortcomings of existing cloud trust models are discussed and their methodologies are analyzed. Further, this chapter explains the common features of existing cloud trust models which are proposed by several researchers. This can be considered as a guideline to determine a proper methodology to propose a unique cloud trust model.

Chapter 2 discusses the methodology of this research in which there are five main phases; analysis phase, matching phase, tracing phase, development and exploitation. The details of these phases are described in this chapter.

Chapter 3 presents an overview of the ISO/IEC standards applicable in cloud computing technology. The purpose of this investigation is to match the relationship between existing system and software quality standards and cloud computing technology to extract standard characteristics. In addition, this chapter compares trust characteristics extracted from the literature to analyzed standards to propose the key trust characteristics in cloud computing. In this chapter we try to present the trust characteristics that are commonly recognized in the literature and by the help of existing ISO/IEC standards, we identify their key sub-characteristics.

Chapter 4 discusses the measures of extracted trust characteristics explained in chapter 3 based on ISO/IEC 25022 and ISO/IEC 25023. These measures are matched with the sub-

characteristics of the specified trust characteristics to contribute a more comprehensive cloud services assessment.

Chapter 5 describes the three cloud service models (IaaS, PaaS and SaaS) and presents their important features. In this chapter IaaS, PaaS and SaaS are compared together and their main similarity and major difference are identified to be able to verify the applicability of trust characteristics measures (in chapter 6) in cloud environments correctly.

Chapter 6 verifies the feasibility of trust characteristics assessment by different categories of cloud users in cloud environments. In this chapter, cloud users are categorized into three groups: individual, organization and cloud service provider (the owner of the cloud service). The details of these categories are discussed in this chapter.

Chapter 7 discusses the main results of the two stages of tracing phase explained in the chapters 5 and 6 and presents the separated models of trust characteristics for the three identified categories of cloud users and the evaluation functions of the proposed measures. These models are the essential parts that later will be applied in certain phases of ED-BeCT.

Chapter 8 explains the phases of the ED-BeCT in detail. In this chapter, the equations for measuring the related trust characteristics, normalization function and other necessary equations are presented.

Chapter 9 presents an example in the form of a case study to describe the phases of the ED-BeCT more clearly. In this chapter the level of trust is calculated based on the hypothetical numbers which are assigned to the trust characteristics measures.

Chapter 10 summarizes the main achievements of this research project, obtained results and the potential benefits that this study will have for the industry. The goal of this chapter is to gather the unique features and significant points of this research project.

Finally, chapter 11 explains the main research steps, describes the significance and the key contributions and explains the limitations and challenges of this research project. Furthermore, some guidelines for further research in the area of trust in cloud computing are provided in this chapter.

Next chapter will elaborate on the first phase of the research methodology that is the analysis phase.

CHAPTER 1

LITERATURE REVIEW

1.1 Introduction

This chapter begins with an analysis of trust characteristics and their measures applied in cloud environments. It will then go on to discuss the applied approach in exploring state-of-the-art on trust models in section 3 and their common features in section 4. Additionally, in section 5 we elaborate on different classifications of trust models and an investigation in the methods of existing cloud trust models is done and an analysis to identify their weaknesses in trust assessment in cloud computing is also presented in section 6. Finally, section 7 summarizes the chapter. We summarized the main research questions that are covered in this chapter in table 1-1.

The articles referred to in this chapter are selected from renowned journals and reputable conferences to identify various aspects of trust and analyze the applied methods in previous models for trust evaluation in cloud environments. Since cloud computing is an evolving paradigm, the articles published in the last decade are mostly considered.

Table 1-1. The main research questions to be covered by investigating in the literature

Research questions	Description
1. What are the important trust characteristics in cloud environments?	Evidently, there is no explanation of trust/trustworthiness in cloud computing standards. Accordingly, the considered trust characteristics in the literature are evaluated based on the various perspectives and different methods. Thus, to propose a comprehensive cloud trust model that is supposed to assess various aspects of trust, specifying the commonly recognized trust characteristics in the literature may be helpful.
2. What are the common features of existing cloud trust models?	The answer of this question might be helpful to develop a trust model with unique features.
3. How the proposed cloud trust models in the literature, can be classified?	Since ED-BeCT is supposed to have unique features, therefore, it is required to specify a category related to its features. Identifying existing categorizations of cloud trust models in the literature may be useful in this regard.
4. How trust is assessed by cloud trust models in the literature?	In the literature, various methods are proposed for assessing trust in cloud computing. In addition, there are several methods to rank cloud services or cloud service providers. These methods can be useful to give insight to develop a comprehensive cloud trust model.
5. What are the main weaknesses in the methodology of the existing cloud trust models?	These weaknesses not only can be a guideline to develop a comprehensive cloud trust model that would be able to overcome the deficiencies of the existing cloud trust models, but also we can partially identify the reason that why there is no broadly acceptable cloud trust model.
6. Why there is not any consensus-based cloud trust model?	In addition to the answer of RQ 5, investigating the literature can provide an appropriate cognition to specify the main gaps in existing trust models. These gaps may be the main reasons to reply to this research question.

1.2 Analyzing trust characteristics applied in cloud environments

There is no doubt that cloud trust model characteristics are mostly based on customer requirements. Furthermore, to the best of our knowledge there is no specific standard or agreed rule to select those characteristics and there is not much research in this area. In all the existing trust models, the main concern is finding the way of calculating trust value rather than standardizing trust characteristics that should be evaluated by the trust model. In this section, we analyzed the trust characteristics which are covered by existing trust models with respect to the criteria defined in the cloud computing standards.

As it was found during the presented analysis, most of the cloud trust models (CTMs) such as the ones proposed in [26-36] consider security as a main trust characteristic. Shaikh et al. in [37] proposed a trust model to evaluate the security strength of cloud computing services. The author in this paper considered nine security characteristics that in his opinion were necessary and sufficient but in reality they were found insufficient based on the dynamic nature of cloud and the criteria in cloud standards. Ghosh et al. in [38] proposed a risk estimation while interacting with a cloud service provider by combining the trustworthiness and competence of a cloud service provider. In addition, there are several papers for evaluation of the CTMs such as [39] that are based on the security while the other aspects of trust are ignored.

Abdallah, E.G, et al. in [40] introduced a trust model for cloud-based applications. This model addressed the four components of security characteristics and mechanism (integrity, availability, privacy and access control) for both man-in-the-middle and man-at-the-end attacks. In this model the author fails to acknowledge the significance of other sub-characteristics of security (such as confidentiality and non-repudiation) as are defined in ISO/IEC 25010 [1].

Singh et al. in [41] proposed a multidimensional trust model that integrates multiple trust characteristics. Cloud Data Trust and Reputation of the Service are considered as two aspects of trust to calculate the trust value. Data Processing, Data Transmission, Data Storage, Data

Privacy and Data Security are covered by the cloud data trust and availability, reliability, Turnaround Time, and Service Use Factors are covered by reputation of the service. However, in [11] trust evaluation is just based on four factors: Availability, Reliability, Turnaround Efficiency and Data Integrity which are considered by the authors the credential characteristics. These models might have been much more convincing if the authors had included all the defined characteristics along with their sub-characteristics in quality standards to evaluate trust.

Li et al. in [42] proposed a trust model to assess servers dynamically and select high-quality cloud services based on the user's requirements. This model integrated multiple trust characteristics and considered the three following characteristics to guarantee service level agreement: security, availability and reliability. However, the considered characteristics may not be adequate to boost users' confidence.

In [43] the trust factors that impact the cloud adoption are introduced as Security, Usability, Reliability, Auditability, Interoperability, Accountability, Controllability and Company Scale. Garg et al. in [44] introduced a framework that measures the quality of the cloud services and proposed a ranking mechanism. This mechanism utilizes AHP (Analytic Hierarchy Process) to rank the cloud services based on multiple-KPIs (Key Performance Indicators). Any number of characteristics can be deployed in AHP but Accountability, Agility, Cost, Performance, Assurance, Usability, Security and Privacy are the suggested mechanism. Obviously, researchers in these articles have not treated trust characteristics in much detail and also the related sub-characteristics identified in quality standards are not taken into consideration.

Trust evaluation in [45] is classified in three groups: direct trust evaluation, indirect trust evaluation and third-party trust evaluation. In each group the trust characteristics such as Time, Quality, Transaction Amount, Transaction number, Geographic Situation, Privacy Protection, Success Rate of Transactions, Operational Stability, Violation Records, Favorable Rate, Page Rank, Quality of Service, Institutional Reputation, The Level of Size and Technology, Security of the Website, The Rate of Customer Complaints are used as indicators. Although, these

indicators can partially be as trust characteristics, the authors have failed to address quality of service characteristics comprehensively.

Habib et al. in [46] proposed a multi-facet trust management (TM) system to help cloud consumers to recognize the differences between a qualified and unqualified quality cloud provider. Customers in this proposed TM system would be able to select their desired characteristics. The system computes a customized trust score of a cloud provider based on these characteristics. Moreover, the CSA/CAI (Cloud Security Alliance/Consensus Assessments Initiative) questionnaire is considered to become a standard. In this model, there is no defined measures for the determined characteristics which can have influence on the model accuracy.

Selvaraj et al. in [47] proposed a dynamic evidence-based trust model. This generalized model did not concentrate on any specific service. The trust model integrated fuzzy inference system and IOWA (Induced Ordered Weighted Averaging Aggregation) operator in order to evaluate the dynamic trust value. In this model author used characteristics selected according to SMI (Service Measurement Index) framework as evidence to evaluate trust. Such approaches, however, are not convincing enough to be considered for trust evaluation, since, the measures of trust evidences are not identified.

Rajendran et al. in [23] proposed a hybrid trust model to evaluate the trustworthiness of a cloud service provider in cloud environment by considering compliance-based and reputation-based trust. In this paper, the reputation was calculated based on user feedback and this feedback was based on QoS characteristics such as Availability, Cost, Customization, Network Speed, Ease of Use, Payment Flexibility, etc. Undeniably, cloud users may have different requirements which can directly affect their feedback. Accordingly, trust evaluation based on user feedback may not be accurate enough to rely on.

Marudhadevi et al. in [48] introduced a trust mining model (TMM) to help users to find a trusted cloud provider while negotiating an SLA. In this work an overall trust value can be

generated by using rough set theory and Bayesian inference. The proposed trust model focused on the techniques of calculating an overall trust degree rather than several trust characteristics that should be covered in the model. Thus, this model may not be able to address different requirements.

Hajizadeh et al. in [49] introduced a trust model by using a behavioral graph and grouping the services. In this model trust evaluation can be done based on four characteristics of availability, reliability, interaction revolution and identity which obviously, are not adequate enough to evaluate various aspects of trust.

Ritu et al. in [50] considered QoS (Quality of Service) parameters such as Turnaround Time, Reliability and Availability and by using fuzzy logic evaluated the trustworthiness of the cloud service provider. The generated result by this model may not be precise as several users requirements are not addressed such as usability. A more comprehensive model would include all the characteristics associated with the QoS characteristics defined in quality standards.

From the literature review it seems that when security is considered to evaluate trust, the other aspects of trust are often ignored or become pale. It is true that in the concept of cloud computing the first concern can be security. Consequently security factors are significant in most of the proposed trust models. But security alone not only is insufficient, but also it should not be the only facet to be considered in order to examine the cloud services and the trustworthiness of the cloud provider.

It is necessary to mention that the terms such as *parameters, features, characteristics* and *factors* are used frequently in different papers. Since the specified term used in the standards is characteristic, this term will be further employed in this research project to facilitate representation in this analysis.

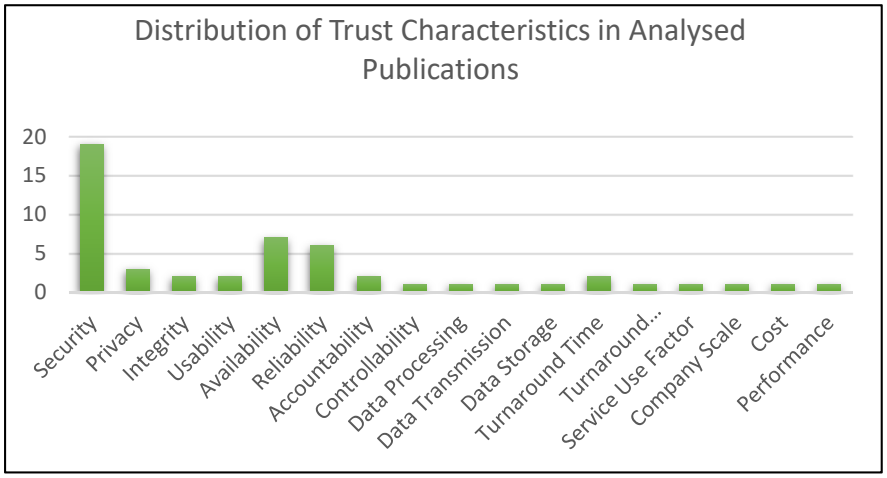

Figure 1-1. Distribution of trust characteristics in the selected papers (Vertical axis represents the number of articles a given characteristic is mentioned in [11, 23, 26-35, 37-50])

Figure 1-1 shows the distribution of trust characteristics in 26 selected papers. The presented trust characteristics in this figure are extracted from journals and conference papers. It needs to mention that the process of extraction has been stopped when the characteristics were repeated in the papers.

1.2.1 The measures of trust characteristics

In this section, we try to specify the general perspectives of various researchers about the trust measures. Thus, the objective of this section is to determine the measures of the characteristics which are considered as trust characteristics in the literature. However, the researchers have not treated these measures in much detail. In other words, so far to the best of our knowledge there is no a single trust model by which the standard measures of the trust characteristics are evaluated.

Lin et al. in [51] proposed a behavior trust model in cloud environments based on an ant colony optimization algorithm (ACO-BTM). Although, this model is able to simulate the trust

relationships behavior between cloud entities, there is no explanation of the measures as well as trust characteristics measurement.

Zhang and Zhou in [52] focused on the trust relationship between users and providers. In this work, the concept of recent trust that aims to measure the most updated values of trust is explained. However, the measures of this measurement are not described.

Noor et al. in [53] presented a platform for credibility-based trust management of cloud services. This platform can provide a crawler for automatic cloud services discovery, an adaptive and robust credibility model to measure the credibility of feedback, and a trust-based recommender to recommend the most trustworthy cloud services to the users. The credibility calculator is designed to measure the credibility of trust feedback according to a set of credibility factors for aggregating the credibility weights. These factors include the users' experience factor and feedback density factor. But the authors in this article fail to fully define the measures based on the related standards.

Although, some research has been carried out on measuring trust in cloud environments, no studies have been found to deal with standard measures of quality characteristics explained in system and software quality standards. Accordingly, this issue can influence on the adaptation of these trust models by cloud service users for assessing trust in cloud environments.

1.2.2 Discussion (Part 1)

As a step towards development of a broadly acceptable cloud trust model, a literature review on trust characteristics was conducted to identify trust characteristics that were proposed in the domain-related research papers. This process is performed by considering the articles published in renown conferences and journals and has been continued up to where the characteristics were repeated.

In addition to the characteristics, we seek to identify the defined measures for these characteristics in the literature. However, no single study exists in which the authors focus on the standard measures (as defined in the related ISO/IEC standards) of trust characteristics and research on the subject has been mostly restricted to the limited criteria that are determined as the general principles.

Overall, these results indicate that :

- There is no consensus-based set of trust characteristics and there is no agreed rule related to the quality of cloud services. An implication of this is the possibility that the users may not be able to assess trust accurately since a considerable part of this assessment should be based on the general knowledge rather than agreed standards;
- Surprisingly, despite the fact that for the quality characteristics there are measures defined in ISO/IEC 25022 and ISO/IEC 25023, the standard measures of quality characteristics are not taken into consideration in the literature. Therefore, it may impact on the trust evaluation negatively.

Thus far, we have explored the literature to find the main characteristics that need to be assessed in trust evaluation process to answer the first fundamental question that is « what » should be evaluated to enable cloud users based on which judge the cloud services. Therein after, the second basic question would be « how » they should be evaluated to produce reliable results which we seek to respond to it in the next section.

1.3 The applied approach in exploring state-of-the-art on trust models

Many scholars hold the view that trust issues in cloud computing are mostly due to the lack of control on data and the inadequate transparency of both, the cloud service providers and the cloud service users [54].

> «A survey conducted among 3000 Cloud consumers from different countries reveals that 84% of Cloud consumers do not

> trust the service providers due to loss of data control on the Cloud platform [54, 55] ».

Migrating data on cloud environments reduces the control that organizations need to have over their data; as data are not stored and processed inside their local environments [54]. In addition, due to the lack of transparency, cloud service users are not aware of the data storage physical location and the security level of the cloud services [54, 56, 57].

To overcome these issues and to assess the provided cloud services, several cloud trust models have been designed. However, there are various problems in existing cloud trust models such as the lack of standardization and interoperability which are the main concerns [54].

> «There is no such generic and comprehensive trust model that can establish trust on all the layers of Cloud services, namely software, platform and infrastructure. Moreover, existing trust models have their limitations in terms of providing essential functionality and security features for trust evaluation [54] ».

Furthermore, due to the rapidly growing interest in trust in cloud computing, the existing literature on trust models is significantly unstructured and incremental [54, 58, 59]. Thus, elaborating on each trust model may not be a precise approach to increase our knowledge and find the gaps in this area and could also be laborious and time consuming as a result of two main reasons :

- It is not possible to analyze all the existing trust models in the literature as there is a great variety of trust models proposed by various researchers with different applicability;
- Since there is not a consensus-based guideline to evaluate trust neither in the literature nor in the related standards, it may not be an accurate method to consider only some models based on certain hypothesis or criteria.

Therefore, as an alternative solution we have tried to : first, explore their common features to gather general knowledge related to these models, second focus on different models classifications to identify different cloud trust models under each category in general and

specify their weaknesses, third analyze their commonly used techniques for trust assessment to give a better understanding of the existing models applicability. This approach is applied to compare and analyze the previous trust solutions and identify their imperfections [54], to develop a comprehensive cloud trust model that would be able to overcome these weaknesses. This approach is depicted in the figure1-2.

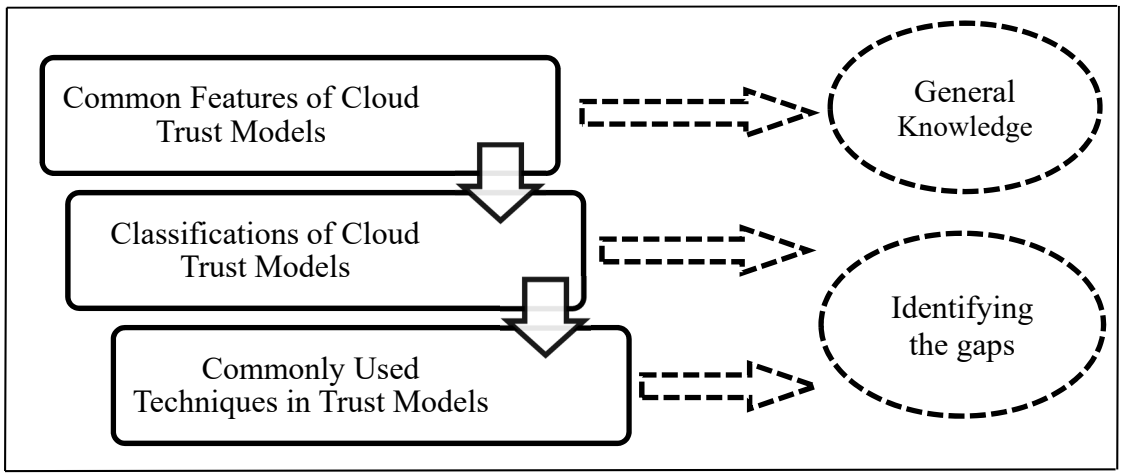

Figure 1-2. The applied approach to explore the literature

As discussed earlier, the first step towards trust evaluation in cloud environments is to specify what should be appraised with what priorities. After this stage, a methodology to assess the identified requirements should be determined. The selected methodology or referred to as model should have potent features to carry out a precise evaluation and at the same time, easy to understand and perform. Having introduced these principles, the following section provides a summary of common features of cloud trust models. Afterward, we will analyze the existing trust model classifications in the literature.

1.4 Common features of cloud trust models

It has commonly been assumed that trust appraisal in cloud environments is a challenging process, since indispensable information related to the quality of cloud services must be gathered from various resources [60].

Trust in the concept of cloud is a three-stages process which has been comprised of user understanding of cloud services and cloud service providers, making decision to whether trust those services or not and adoption behavior of cloud such as action of purchasing [61]. Considering this process, there are three potential references for trust [62] :

- Trust in the cloud provider: this part of trust associates with establishing an agreement with cloud service users such as Service Level Agreement (SLA), that clarifies the obligations for quality such as the value of response time;
- Trust in the cloud service: this part of trust is related to the competence of the service in order to perform the required tasks. This may be assessed according to cloud user satisfaction and the previous experience of users with the service;
- Trust in the cloud itself: It is commonly considered as the most difficult part, as it is related to the effort for convincing users to adopt cloud as a technology and migrate their data to the cloud environments.

Distinguishing these different parts of trust clearly is a critical phase in building trust in the interaction with the cloud technology [63].

Recent research has suggested various methods to facilitate the evaluation of trust and improve the user perceptions of cloud services. These perceptions of the other party that can affect trust directly are known as trustworthiness which can be organized in three categories [62, 64] :

- The first category is related to the functional aspects of cloud services which enable users to fulfill their goals;
- The second category is related to the feeling that there is support for using cloud services;
- The third category is based on beliefs that the service would be reliable and its functions would be predictable.

Cloud service providers that can successfully introduce the characteristics for trustworthiness to the cloud service users, build the knowledge-based trust with their users, which is considered as the first step of trust in reality [62]. For the next step of trust in reality, the cloud trust models can be very helpful and act as an assistant for cloud service users. The following part of this chapter moves on to describe in greater detail the common features of cloud trust models which are mostly extracted from [65].

The aim of each trust model is to find how trust can be perceived and computed in a setting which is computational. In this setting, there must be two (or more) entities which have interaction in different ways. In each trust setting, an entity plays one or more role(s). Briefly, these roles are 1) the trustor that is the entity which sets the trust, and 2) the trustee that is the entity that the trust can be set on that. There might be the other roles that relies on the model application and the complexity of that. For example, an entity can act as a witness in the way that its opinion based on its previous interaction with an entity can be used in the model for evaluating trust. For the trustors and trustees, there are some specific roles such as a requester of the service, the provider of the service and a third party that is trusted and can issue certifications or collect the other entities feedback for computing the score of the service reputation. A trust relationship can be built once a trustor and a trustee exist. To evaluate a trust relationship, a trust value can be calculated.

There is a purpose for establishing a trust relationship in each trust model. These purposes are based on the fact that trust depends on the context which is the most important aspects of trust. As it can affect the other concepts such as the type of an entity and its role. In addition to the context, there are some other influential factors for trust. The trustee's subjective and objective properties and the trustor's subjective and objective properties [65].

Basically, a cloud trust model has different components such as cloud service discovery, trust metric selection and measurement, trust assessment, trust evaluation [66] and finally, according to this evaluation a trust certificate can be issued for the evaluated cloud service. This certificate can be helpful for the selection of one or more cloud service provider(s) who

offer the services which can fulfill the maximum set of functional and non-functional user's requirements.

On the other hand, cloud trust models would depend on the type of stakeholder which are discussed in the following three scenarios [8] :

- A cloud service user (trustor) wishes to establish trust with the trustee (cloud infrastructure and cloud provider) for providing services S, and to impel a policy P which is agreed by both trustor and trustee, when the both sides have the behavior B;
- In this scenario, the trustor can be a cloud provider or two or more clouds that have collaboration together, wish to establish trust with the trustee (cloud infrastructure or the cooperating cloud provider and its infrastructure) for providing services S, and to impel a policy P which is agreed by both trustor and trustee, when the both sides have the behavior B;
- In this scenario, the trustor is the customer of a cloud user (direct and indirect users), and this customer wishes to establish trust with the trustee (cloud user resources at the cloud infrastructure) for providing services S, and to impel a policy P which is agreed by both trustor and trustee, when the both sides have the behavior B.

Moreover, cloud trust models might comply several modeling methods, such as mathematics, linguistic, graphic and apply several approaches such as dynamic or static approaches [65]. These methods are considered to classify different trust models in the literature. We discuss the details of these classifications in the next section.

1.5 Classifying cloud trust models

Before delving into the details of cloud trust model classifications, it is essential to clarify the concept of trust and trust models in cloud computing. Although, there are multiple definitions of trust in the literature (such as [67] in which trust has a hybrid notion that combines hard and soft trust mechanisms to validate the trust properties [36]) and there is little consensus about

what trust actually means [68], Kanwal et al. in [54] mapped the definition of trust from [69] to the cloud perspective as :

> «the expectation of a Cloud consumer regarding the actions and behavior of a cloud service provider (CSP) that will affect the consumer's choice in the selection of a CSP »

Therefore, according to Kanwal the results of trust assessement have direct effect on the selection of a cloud service provider. In this regard, applying a proper cloud trust model is fundamental to make this decision. Furthermore, the author in [54] defines trust models as different mechanisms, techniques and protocols proposed in the literature to evaluate the trustworthiness of the cloud service providers. Besides, a trust model refers to a coded implementation that depends on the concepts of trust to identify a trust value related to a cloud service provider and according to which there are restricted and controlled interactions with that cloud service provider [54, 70-72]. Accordingly, to analyze the current state of the art, the existing trust models (e.g. [40, 41, 73]) need to be classified in a rigorous way.

Different methods have been proposed to classify the existing cloud trust models. For example, trust mechanisms in [74] are classified in four groups :

- static dynamic : an example for this mechanism is presented in [37]. This mechanism is identified as cloud service features and specifications which are considered to assess the trust value as static trust. This trust value is according to the user experience and interactions over a time frame. In addition, in this work [37] a refined set of parameters are calculated to assess the trust dynamically;

- direct indirect : in the work presented in [75], there are several factors that the direct trust depends on; a-historical evaluation, b-transaction time, c-transaction amount. In addition, in [76] direct indirect mechanism refers to the information sources that are used in trust model (i.e. direct experience and direct evaluation - indirect experience that is the value of trust obtained by the other trustworthy groups from a cloud service provider). However,

direct evaluations will require precise methods or tools and in indirect experience also determining a trustworthy group needs further research to specify the essential criteria for their selection;

- centralized-distributed : as explained in [74] and [77], in centralized mechanisms, the entire architecture is controlled by a single entity. In case of demanding a service by an entity, a request will be sent to the server by the entity, then the server finds proper resources and assigns them to the entity that needs it [74, 78]. The centralized mechanisms in cloud computing have scalability problems [74]. The examples of this mechanism are presented in [21, 79]. In distributed mechanisms, the control operations are performed by the network components. These components can cooperate to perform the control mechanisms flexibly [74, 77]. The distributed mechanisms require extra network synchronization prices [74]. The examples of this mechanism are presented in [80, 81];

- proactive-reactive-periodic : surprisingly, for this group of cloud trust models which is proposed in [74], there is not any example or explanation in the literature.

On the other hand, Moyano et al. in [65] determined two types of cloud trust models :

- **Decision models :** These models provide flexible decisions of access control and simplify the two steps of authentication and authorization processes into one step of trust decision. The examples of this category are **policy models** and **negotiation models** :

  - **Policy models :** It is clear from their name that policy models apply policies, which identify the main conditions under which access to a resource is provided. These conditions are generally considered as credentials or signed statements that ensure an entity is which it asserts to be, or that is a group member. They are associated with the notions of policies and credentials, restricting the access to resources under specified policies that determine which credentials are required to access them. An example of these models under this category can be found in [82];

- **Negotiation models :** They are the other type of trust decision models. These models by adding a protocol which is called negotiation strategy in case of performing a step-by-step negotiation-driven exchange of credentials and policies between two entities, help them to decide whether to trust each other or not. This strategy protects the privacy of the entities since policies and credentials are disclosed only if it is needed. An example of these models under this category can be found in [48].

As each cloud provider has their own policy for restricting the access to resources and negotiation strategy, it seems that the major drawback of the decision models may be that they are applied to evaluate the internal policy of the cloud service providers rather than trust in cloud environments. Thus, the results of these models might not be reliable and comprehensive adequately.

- **Evaluation models :** These models refer to the computational trust that is proposed by Marsh [83]. As explained in this article, they evaluate the reliability (or other related attributes) of an entity by assessing defined features that impact on trust (**behavior models**), or by distributing trust information across trust chains (**propagation models**). Reputation models are an essential subtype of behavior models, and recommendation models are an important subtype of propagation model. In the following, the behavior models and propagation models are briefly described :

  - **Behavior models :** these models have three phases. In the first phase, a bootstrapping phase may be needed to allocate initial trust values to the system entities. The concept of trust propensity is associated with the bootstrapping phase and refers to the tendency of the model to high or low trust values primitively;

  In the second phase, monitoring will be performed to monitor a parameter or a set of parameters. In the third phase, an evaluation process is performed to determine values to the observed qualities and to combine them into a final score of trust;

In behavior trust, each trust relationship has a value of trust that refers to what extent the trustor can trust the trustee. This trust value may be either uni-dimensional or multi-dimensional and is computed by using a trust evaluation process and considering trust metrics. Trust metrics apply variables, such as risk or utility, and aggregate them in order to determine a final score for the measured attribute(s). Trust metrics use computation engines, which might include simple summation or average engines, continuous engines, discrete engines, belief engines, Bayesian engines, fuzzy engines or flow engines. The source of information for the metric may be provided by direct experience which may be direct interaction or direct observation, sociological and psychological parameters. This is one of the significant advantages of these models since the source of information for the metric will be obtained dynamically and provide a high compliance with dynamic nature of cloud computing to assess trust. Thus considering this process, behavior trust models may be helpful in the development of a comprehensive model and in comparison with the other classifications, the models under this category are able to produce more comprehensive results, as they consider the behavior of the services dynamically. However, the studies in this field of trust have mostly focused on the user behavior rather than cloud services behavior. An example of these models under this category can be found in [84];

- **Propagation models** : these models consider that various trust relationships have already been created and quantified. However, this is not at all times the case. Their goal is to establish new trust relationships by propagating the information of trust values to the other entities. An example of these models under this category can be found in [85].

In the same research field, Deshpande and Ingle in [86] adopt the classification of trust models in [65] and extend it by linking to various trust evaluation techniques. Figure 1-3 shows this classification presented in [86].

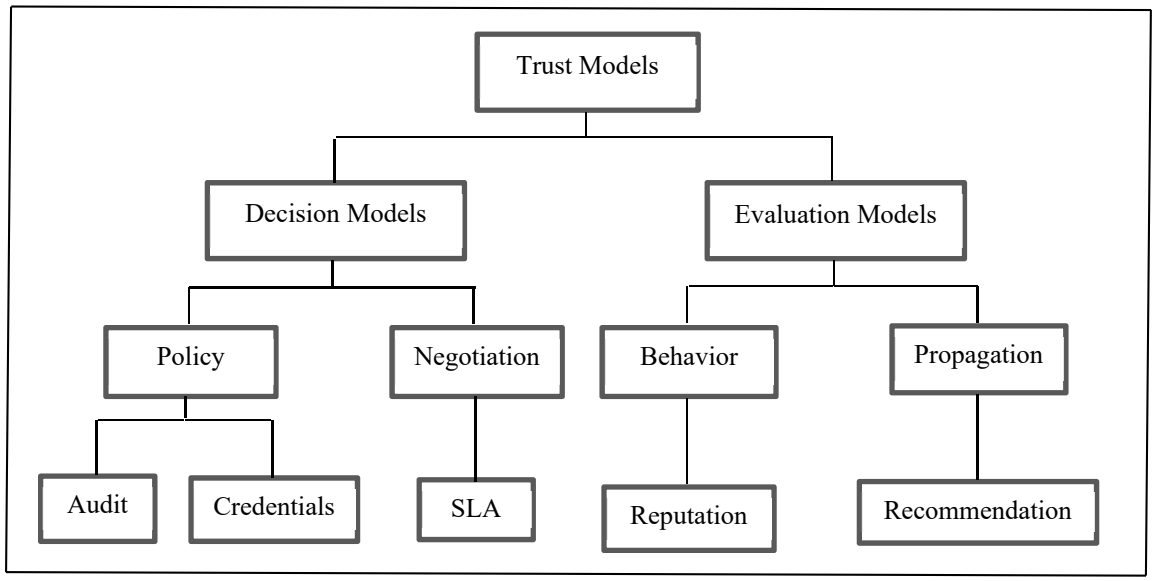

Figure 1-3. Classification of trust models
Taken from [86]

In contrast to Deshpande and Ingle, Rawashdeh et al. in [87] proposed a new classification. The authors in this article classified the trust models based on their applied technologies and research trends to establish trust in diverse cloud service providers. As explained in [87], some examples of applied technologies are; SLA (Service Level Agreement), auditing, measurements and self-assessment questionnaires (CSA) [88], while in the literature there are several trust models that cannot be recognized under the presented categories, (such as Context-Aware Multifaceted Trust Framework presented in [60]). This classification is depicted in figure 1-4.

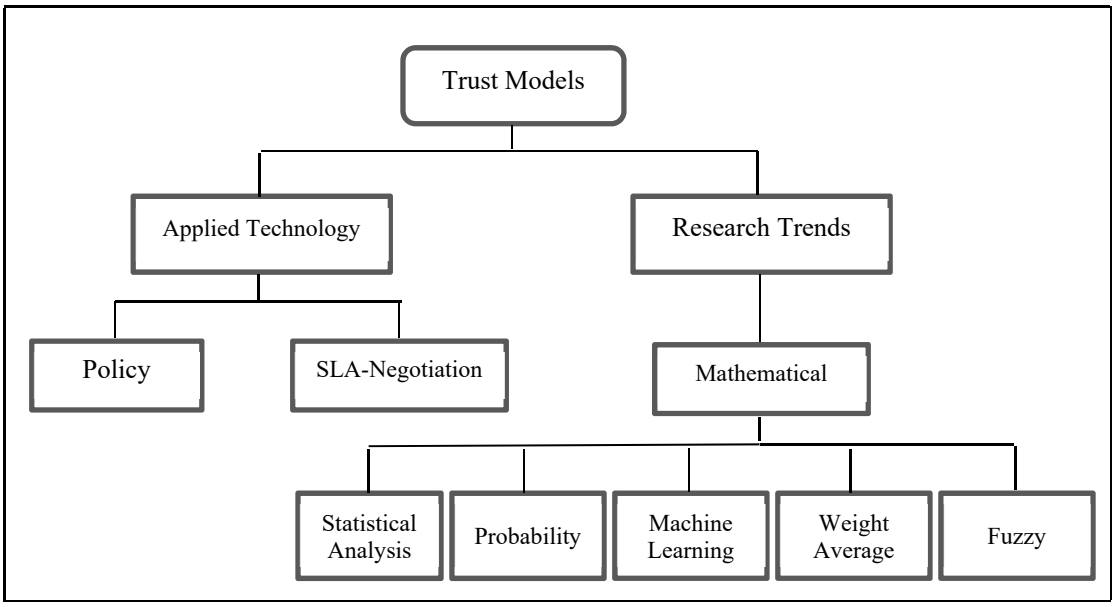

Figure 1-4. Classification of trust models

Taken from [87]

The components of each category presented in figure 1-4 are explained as follows [87] :

Applied technology

- Policy models : To control the access to different resources, some policies and credentials are used. The policies introduce the necessary permissions to access various resources as auditing and credentials;
- SLA-Negotiation models : Trust will be established based on the agreements between cloud service providers and cloud service customers [89].

Research trends trust models

These trust models based on mathematical methods [54] can be applied in many common trust modes that are discussed as follows :

- Weighted Average : These types of trust models are the trust values that can be formed on the weighted average. The trust evaluation formation of the overall views related to the different aspects, can be divided into the local trust models and global trust models;
- Probability : These models mostly use the estimation of the maximum probability and various calculation methods are applied to compute the trust value;
- Fuzzy Logic : The author in this article argues that membership in the theory is considered as fuzzy and the body pertains to a reliable group. Based on the fuzzy rules on the evaluated fuzzy data, the trust value will be determined [90]. The fuzzy process can be divided into three steps : fuzzification, fuzzy inference and de-fuzzification;
- Statistical Analysis : In these models, based on the various application contexts by considering more than one dimension related to the trust (i.e. historical information, contextual information, and reputation information) the high-confidence trust relationship will be predicted;
- Machine Learning : These models in this article are categorised into two groups; supervision trust forecasting methods – unsupervised trust forecasting methods. The idea is by using a machine for learning methods, the dynamic rules are generated, then by combining a fuzzy reasoning and rule-based, the level of trust will be determined.

Noor and Sheng in [91] argue that various trust management techniques can be classified in four different groups which are discussed as follows :

- Policy as a Trust Management Technique (PocT) : This is one of the most common methods to establish trust among entities in cloud environments. PocT applies a set of policies and by considering various rules to control authorization levels, determines a minimum trust threshold to authorize access. The trust threshold is based on the trust results or the credentials.

  Several approaches can be applied for the trust results-based threshold. For example, the monitoring and auditing approach, the entity’s credibility approach, and the feedback credibility approach.

- Recommendation as a Trust Management Technique (RecT) : This is another popular technique in trust managements which benefits from participants' knowledge of the trusted entities, particularly regarding to the fact that the entity knows the source of trust feedback. There are various forms for the recommendation technique, the explicit recommendation and the transitive recommendation are the two examples of this technique. As explained in this article :

> « An explicit recommendation happens when a cloud service consumer clearly recommends a certain cloud service to her well-established and trusted relations (e.g. friends). A transitive recommendation happens, on the other hand, when a cloud service consumer trusts a certain cloud service because at least one of her trusted relations trust the service. »

- Reputation as a Trust Management Technique (RepT) : This technique is essential since the feedback of the different cloud service consumers can significantly impact on the reputation of a specific cloud service;
- Prediction as a Trust Management Technique (PrdT) : This technique is advantageous particularly in case of having no prior information related to the cloud service interactions.

Basically, the authors in [91] tried to clarify various methods and techniques which are used in the literature to assess trust in cloud computing and based on these techniques, a classification of trust models is proposed. However, it is worthy to note that as the dynamic aspect of trust in this proposed classification is not considered, to conduct an analysis of existing cloud trust models, this classification may not be comprehensive. However, this view is supported by Khan et al. in [92] and proposed a more complete classification of the existing trust management techniques:

- Service Level Agreement (SLA) based trust management technique- In this technique the cloud service consumers assess the conformity of the cloud service providers to the SLA parameters to specify the trustworthiness of the cloud service provider;
- Recommendation based Trust Management Technique;

- Reputation based Trust Management Technique;
- Prediction based Trust Management Technique;
- Policy based Trust Management Technique.

On the other hand, Kassabi and Serhani in [93] conducted a survey on the existing reputation trust models. As explained in this article, reputation and the concept of trustworthiness are related, although they are different [68, 93]. Reputation in this work is defined as :

> « What is generally said or believed about a person's or thing's character or standing. »

Also, the quality of service provisioning related to a cloud service provider which is perceived by its users, will specify its reputation [93]. It is argued in this article that having an acceptable reputation will lead to trust, but as the nature of trust is subjective, trust can also be considered for a cloud service provider according to a pleasant self-experience even with having a bad reputation. Nonetheless, trust is generally determined by reputation, when the self-experience does not exist.

The authors in this work [93] explained that a reputation based trust model depends on the opinions and previous experiences of other users with the cloud service providers. The reputation based trust models in this article based on the type of quality characteristics considered to calculate the trust score, are classified into two groups of service oriented and resource-oriented models :

- Service-oriented models : These models evaluate trust based on the guaranteed quality of services of the cloud service provider;
- Resource oriented models : These models rely on the quality resources which are provided by the cloud and the availability for computing the score of trust.

According to Jøsang et al. in [68] and Resnick et al. in [94], the three properties of reputation systems are as follows :

- Entities should be lived during a long time, in this way with each interaction, an expectation of future interactions always exists;
- Ratings of current interactions are obtained and distributed;
- Ratings of previous interactions can steer decisions about current interactions.

The authors in [68, 94] explained these three properties as follows :

The first property that is durability of agents, is associated with the situation in which it may not be possible or easy for an agent to modify identity or pseudonym with the intent of eliminating the connection to its past behavior. The second property depends on the protocol for ratings, which is not basically a problem for centralised systems, but creates a major challenge for distributed systems. This property also depends on the inclination of participants for providing ratings, for which there must be some forms of motivation. The third property depends on the usability of the reputation system, and the way with which people and systems respond to it.

Conversely, the major difference between trust and reputation systems can be explained as follows [68] :

- The trust systems generate a score that refers to the relying party's subjective view of the trustworthiness of an entity, while reputation systems produce the reputation score of an entity as seen by the whole community;
- In trust systems transitivity is an explicit component, while in reputation systems usually transitivity is the only component that implicitly will be taken into consideration;
- In trust systems usually subjective and general measures of trust are used as input, while in reputation systems, the information or ratings of specific objective (e.g. transactions) are considered as input.

As Jøsang et al. discuss in [68], this is possible that trust systems incorporate the components of reputation systems and vice versa, but the way of classifying a given system is not always clear. The important point in reputation based trust models is that the benevolence and the good intention of the entity which is supposed to rate the cloud services is essential. Otherwise, the result of trust evaluation may not be reliable.

A broader perspective has been adopted by Kanwal et al. in [54, 95] who classify trust models into five classes on the basis of their various approaches for computing the score of trust. The proposed classification includes agreement based, certificate based, feedback based, domain-based and subjective trust models. In addition, the authors in [95] proposed seven assessment criteria to evaluate the trust models in each category. These assessment criteria are data integrity, data control and ownership, process execution control, quality of service (QoS) characteristics, detection of untrusted entities, dynamic trust update and logging, and model complexity. In the following, a summarized description of these criteria are given (the indicated level of High, Medium and Low are based on the level of compliance related to each feature) [95] :

- Data integrity : data integrity is related to the security of users' data against malicious threats or unwanted modifications. Trust models need to assure data integrity for security of users' data. In this analysis, High level is considered for the models that ensure evaluate this criterion through various encryption techniques. Medium level is considered for the trust models that assess this criterion through SLAs or certificates issued to CSP by a third party. Low level for trust models is considered when there is no technique to guarantee this criterion on cloud;
- Data control and ownership : This is related to having authorized and authenticated access to stored data on cloud under the full control of users [56]. The trust models that are able to assess this criterion through access control policies, trust tickets or trusted platform module which is managed by users, are indicated as high level. Low level is considered if a model does not evaluate this criterion;

- Process execution control : This is related to the performed control of the computational activities on deployed applications on cloud [56]. For example, all the computational activities for an image searching application deployed on cloud are under control of CSP in SaaS (such as image temporary storage, image update and image removal). However, as explained in [96], a user can control the tasks through remote access control or virtual private networks that augment the trust level of CSP. A trust model which is able to assess this criterion via virtual trust infrastructures or trusted platform module is considered with high level for this criterion. Low level is considered for the models without the process for assessing this criterion;
- Quality of service (QoS) characteristics : This criterion is related to direct measurements of quality characteristics such as response time or throughput or indirect measurements of these characteristics by considering the collected feedback and opinions. High level is considered for the model with direct measurements techniques through various formulas, medium level is considered for the models with indirect measurement techniques through feedback or opinions, and low level is assigned for the models without measurement techniques;
- Detection of untrusted entities : The trust models that contain the mechanisms to detect the entities providing fallacious feedback in cloud. High level is considered for the models that completely detect untrusted entities through various techniques such as applying majority consensus. Medium level is considered for the models that can partially detect the untrusted entities via defining the static threshold value below which the entities are considered in untrusted zone. In this mechanism due to this static threshold, certain malicious entities in dynamic cloud environments cannot be identified. The models without this feature are specified as low level;
- Dynamic trust update and logging : Trust models need to support logging of transactions between cloud users and cloud providers to boost the level of users' confidence. In this analysis, this feature is considered as high level for the models that support logging by monitoring the history of transaction and updates the score of trust dynamically through update policies, time decay function or history records in logs. Medium level is considered for the models that support none of the mentioned features since they provide the dynamic

trust update but do not encounter any mechanism for logging of transactions or vice versa. Low level is considered for the models without the both logging and dynamic trust update parameters;

- Model complexity : This is related to the context of ease in practical scenarios with fewer configuration phases. Thus, the trust models with least complexity level are more practical and useful. High level for model complexity is considered for the models that are complicated to apply and implement as a result of their complex functionality or configurations. Medium level is considered for the models with the ease in applicability of model but with the complex mathematical algorithms. Low level for this feature is considered for the models with simple configurations or mathematical algorithms.

On the one hand, the proposed classification in [54, 95] has two attractive features: 1- the authors defined certain assessment criteria for the proposed classification which in the previous mentioned classifications do not exist, 2- the authors in this classification tried to cover a broader view of trust models in comparison with the previous mentioned classifications.

On the other hand, the defined assessment criteria are not comprehensive and the essential principles such as the role of quality standards (e.g. ISO/IEC 25010) and cloud computing standards (i.e. ISO/IEC 17788) in trust assessment by the models and evaluating the standard measures related to the QoS characteristics are not considered. This classification is presented in figure 1-5 and briefly described as follows [54, 95] :

- **Agreement-based trust models**

  General description of this category

  In this category to establish trust, it is necessary to sign an agreement by cloud service providers (CSPs) to collaborate and deliver different services to cloud service users (CSUs). In the first step of assessing trust, the CSU identifies its diverse security and QoS

requirements to the module of trust assessment. This module can create and negotiate the contract with the CSP. SLA or service practice statement can be used as this agreement. In the second step, the module of trust assessment sends a request of agreement negotiation to the CSP along with the CSU's required characteristics. Then, the module of contract characteristics monitoring exchanges this agreement with the CSU to establish trust between them [54, 89, 95, 97]. In this category of trust models, to establish trust, several security concerns and QoS characteristics are added in the agreements [95]. SLA-based and security aware models are the two subtype of this category.

SLA-Based Trust Model

As explained in [89, 95], these models include the main components of cloud service discovery, SLA-agent and cloud consumer module.There are three stages in the trust assessment process : 1- the customer will discover and select cloud service providers based on the required functional characteristics, 2- the SLA-agent will design and monitor the SLA parameters and prepare a report for the trusted cloud providers. 3- the trust management module is responsible for calculation of the final trust value for a specified provider by applying direct experience with providers, opinion of external providers and report provided by SLA-agent.

- **Analysis of SLA-based models [95]:** The SLA-management module in SLA-agent creates and negotiates the policies of access control related to the data stored on cloud, hence, the data control and ownership to users are guaranteed. However, these models do not contain any method to ensure the process execution control. In addition, business activity monitoring module is responsible for monitoring and maintaining the details of users' transactions but due to static SLAs, these models do not support the dynamic trust update. Also, SLA agent is responsible for designing and selecting the required characteristics for SLAs, whereas the tasks are managed and created by CSP to introduce simplicity in applicability, hence these models support low complexity. Table 1-2

summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-2. Strengths and weaknesses of SLA-based trust models

Adapted from [95]

Strengths of SLA-Based Trust Models	Weaknesses of SLA-Based Trust Models
• Assuring the data integrity • Guaranteeing the data control and ownership • Assuring the QoS characteristics • Detection of untrusted entities • Low complexity	• Not assuring the process execution control • Not supporting the dynamic trust update

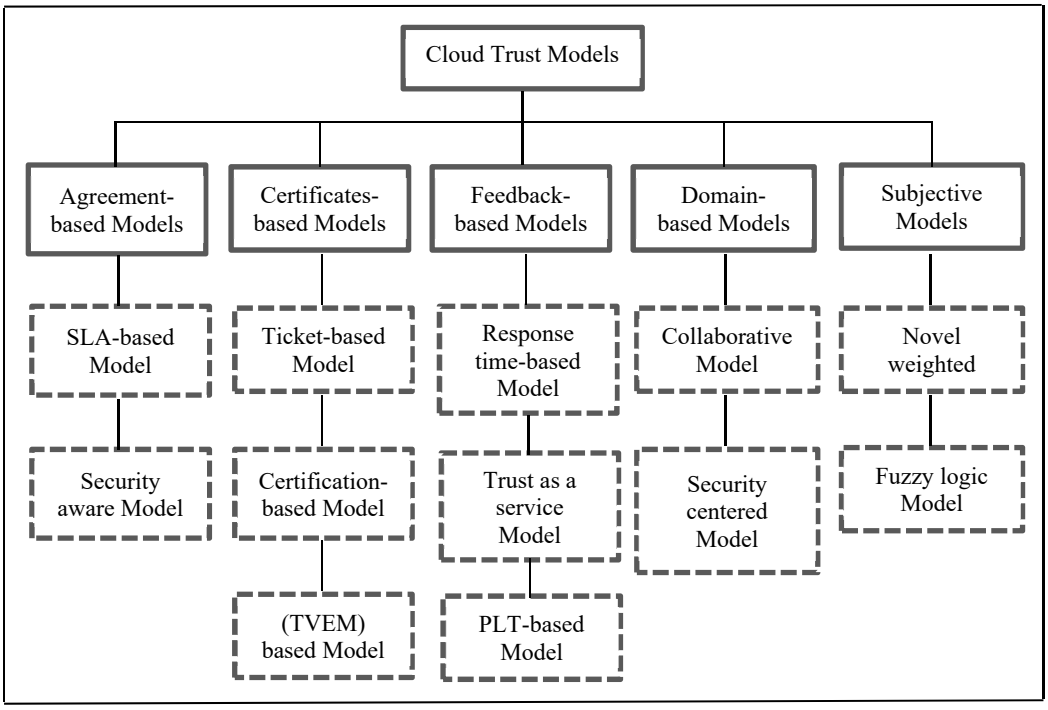

Figure 1-5. Cloud trust models classification

Taken from [54, 95]

Security aware models

The general idea of these models are discussed in [95, 97] which contains two hierarchy levels for trust : internal trust layer and contracted trust layer. If the underlying operations are under the internal control of organization, the internal trust layer will be established on cloud platform. The internal trust can be achieved through trusted platform module which is responsible for assessing the underlying configurations of cloud and declaration of identity and main management under the full control of organizations. The contracted trust is related to service policy statement in such a way that provider will enter into this trust layer through negotiation related to the required security and QoS.

- **Analysis of security aware models [95]** : These models under this category are able to ensure data integrity on cloud through revocation and management of encryption keys under user control. In addition, the data control and ownership will be assessed by negotiated identity provisioning and access control. In general, detection of untrusted entities is supported by assessing platform configurations of CSP. Although, these models are able to ensure the process execution control, they cannot provide any mechanism to assess and ensure the QoS characteristics. Moreover, these models are not able to retain the logs of users' transactions. Accordingly, there is not any mechanism to simplify the dynamic trust update. The service agreements in practical scenarios would be easy to use, without any involvement of complex configurations. Thus they support low complexity. Table 1-3 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-3. Strengths and weaknesses of security aware trust models

Adapted from [95]

Strengths of Security Aware Trust Models	Weaknesses of Security Aware Trust Models
• Assuring the data integrity • Guaranteeing the data control and ownership • Assuring the process execution control • Detection of untrusted entities • Low complexity	• Not assuring the QoS characteristics • Not supporting the dynamic trust update

- **Certificate/secret keys-based trust models**

General description of this category

Trust is established by certificates, trust tickets (TTs) and endorsement keys which are provided by the certificate authority. The main practical solution to establish trust is security certificates for software, platform and infrastructure services. TTs are issued to assure the integrity and confidentiality of data in cloud environments and to boost users' confidence, [98]. To assure the users' control over its data in cloud, several certificates and secret keys can be used [99, 100]. The trust models based on trusted platfom module (TPM) endorsement keys to assess the configurations and measurements to establish trust, also can be included in this category. Ticket based trust model and Trusted Virtual Environment Module (TVEM) based trust model are the subtypes of certificate/secret keys-based trust models.

Ticket based trust model

These models in [98] are proposed to establish trust on cloud service providers. The owner of data will encrypt the data by applying the secret keys which are shared between owner of data and users, and will send the encrypted data to the provider with the capability lists

of already registered users such as usersid, dataid and access rights. In trust ticket generation protocol, a new user will register itself by data owner and sending the needed credentials. Then, the data owner can produce the capability list and trust ticket (TT). In trust ticket deployment protocol, the encrypted TT is sent by the user to the provider to provide access to the stored data on cloud. After verification of the credentials and trust ticket, the provider will send the encrypted data to the user.

- **Analysis of ticket-based trust model [95]:** These models have the high level of data integrity through encryption techniques and ensure data control and ownership provided by CSP. Further, before sending the encrypted data to the user, the capability lists and TT will be verified by CSP to ensure the detection of untrusted entities. These models cannot assure the process execution control and QoS characteristics provided by the cloud service provider and do not encounter any logging feature. In addition, they are not able to support the dynamic trust update. The capability lists and TT have easy practical applicability, while managing and distributing a unique secret key for each user have complexity up to medium level. Table 1-4 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-4. Strengths and weaknesses of ticket based trust models

Adapted from [95]

Strengths of Ticket Based Trust Models	**Weaknesses of Ticket Based Trust Models**
• Assuring the data integrity • Guaranteeing the data control and ownership • Detection of untrusted entities	• Not assuring the QoS characteristics • Not assuring the process execution control • Not supporting the dynamic trust update • Complexity

Trusted Virtual Environment Module (TVEM) based trust model [95, 100] :

These models have a TVEM manager, a virtual trust network (VTN) and a TVEM factory (TF). TVEM is a software module which is generated in TF having the data owner control. TF for data owner can generate TVEMs, root VTN master key, VTN certificate and trusted environment key (TEK). VTN root certificate key is produced by TF module that is a master key to secure and revoke the other keys for each VTN. TEK is the endorsement key for TVEM rooted in host cloud platform and VTN is used for establishing trust in virtually established cloud environment. TVEM measures the cloud platform trust across the core root of trust for measuring (CRTM) and trusted computing base (TCB) that includes the BIOS and various configurations.

- **Analysis of Trusted Virtual Environment Module (TVEM) based trust model (TVEM) [95] :** To guarantee the data integrity, the TEK and VTN are applied. To ensure data control and ownership, TCB measure the configurations and Virtual Environment Configuration Registers (VECR) assesses the virtual environment policies. All the computational activities are under control of VTN. Also VTN ensures the process execution control. CRTM and TCB are provided to be helpful to detect untrusted entities. But these models cannot assure the QoS characteristics of the cloud services and there is not any parameter to keep the virtual environment logs. The TVEM manager on cloud platform is able to measure the configurations of CRTM and TCB to ensure the dynamic trust update. However, creation, management and migration of TVEM, VTN certificates and TEKs are complicated to apply. Table 1-5 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-5. Strengths and weaknesses of TVEM based trust models

Adapted from [95]

Strengths of TVEM Based Trust Models	Weaknesses of TVEM Based Trust Models
• Assuring the data integrity • Guaranteeing the data control and ownership • Assuring the process execution control • Supporting the dynamic trust update • Detection of untrusted entities	• Not assuring the QoS characteristics • Complexity

- **Feedback-based trust models**

General description of this category

The trust models in this category collect opinions and feedback of the users to assess the trust. In the first step, different CSPs are enrolled with the trust model through the module of service registery [88, 101, 102]. Then, the module of feedback collects and manages the users' feedback based on the various QoS and security characteristics which are offered by the registered CSPs. The module of trust assessment formulates the score of trust for CSPs based on the collected feedback. Then, the CSUs can send the request to the module of trust assessment to obtain the score of trust for the required CSP. Finally, the score of trust will be returned to the user. Response time based models, trust as a service model and prepositional logic terms (PLT) based trust model are the subtypes of this category.

Response time based models

Although there is no explanation about this subtype of trust models in general, it seems that these models deal with the amount of time that is lasted to respond to a users' requirements.

In addition these models can contain the mechanisms to evaluate QoS characteristics (e.g. [103]).

- **Analysis of response time based models [103]** : In these models, there are mechanisms to partially evaluate QoS characteristics which is one of the essential responsibilities of cloud trust models and ensure data integrity and they have no mechanism to ensure data control and ownership as well as detection of the untrusted entities and the process execution control. However, they ensure the dynamic trust update and represent low complexity in application. Table 1-6 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-6. Strengths and weaknesses of response time based trust models
Adapted from [103]

Strengths of Response Time Based Trust Models	Weaknesses of Response Time Based Trust Models
• Assuring the data integrity • Supporting the dynamic trust update • Assuring the QoS characteristics • Low complexity	• Not guaranteeing the data control and ownership • Not assuring the process execution control • No detection of untrusted entities

Trust as a service models [95, 102] :

The framework of these models includes a registery service component and three main layers (i.e. provider layer, trust management service layer (TMSL) and cloud user layer). The TMSL collects the user's feedback about various services and evaluates the trust values according to the collected feedback. The trust feedback will be collected in the form of history record in which the identity of providers, identity of user who is the owner of the feedback, a set of feedback and weighted credibility feedback are included. The

cloud user's capability and majority consensus are the applied methods to specify erroneous feedback.

- **Analysis of trust as a service model [95] :** In these models, the integrity of data in cloud environments cannot be ensured. However, the received feedback from users are helpful for ensuring QoS characteristics provided by CSPs. Majority consensus and cloud user's capability can provide dynamic credibility to ensure detection of the untrusted entities. In addition, there is not any method for process execution control of users. They do not ensure that all the accesses to data can be under full control of cloud users, so there is no guarantee for data ownership and control. They provide a medium level of support for dynamic trust update and logging features since there is no mechanism for retaining the transaction logs. Due to the deployment of the trust management layer at separate infrastructure between user and cloud layers, the applicability of this model includes complexity. Table 1-7 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-7. Strengths and weaknesses of trust as a service models

Adapted from [95]

Strengths of Trust as a Service Models	Weaknesses of Trust as a Service Models
• Supporting the dynamic trust update • Assuring the QoS characteristics • Detection of untrusted entities	• Not assuring the data integrity • Not guaranteeing the data control and ownership • Not assuring the process execution control • Complexity

Prepositional logic terms (PLT) based trust model [46, 95] :

These models have the main modules included Consensus Assessment Initiative Questionnaire (CAIQ) Engine, Registration Manager (RM), Trust Semantic Engine (TSE),

Trust Computation Engine (TCE), Trust Manager (TMg) and Trust Update Engine (TUE). TSE is responsible for configuring various formations of prepositional logic terms (PLT) that present the trustworthiness of the provider behavior related to the specified characteristics. While TUE is applied to accumulate feedback and update trust values.

- **Analysis of prepositional logic terms (PLT) based trust model [95]:** The questions designed by CAIQ engine about different encryption techniques applied by CSPs to evaluate data integrity. Also TUE collects users' feedback to ensure various encryption methods provided by a CSP. The data from CAIQ engine and TUE will be combined through PLTs to evaluate encryption techniques, thus ensuring the data integrity up to medium level. But in these models there is not any mechanism to establish access control policies to ensure the data control and ownership or users. These models cannot control the process execution offered by CSP for the users. Different feedback related to the QoS characteristics are combined through PLTs and the assessed value of trust is depicted to the user. Hence these models support partially the assessment of QoS characteristics. Further, consensus and discounting operators can gather the opinions and allocate weights of credibility to these opinions thus, they can detect the untrusted entities. The PLTs configurations and formulations present medium level of complexity. However, they do not have any mechanism for logging features and dynamic trust update. Table 1-8 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-8. Strengths and weaknesses of PLT based trust models

Adapted from [95]

Strengths of PLT Based Trust Models	Weaknesses of PLT Based Trust Models
• Assuring the data integrity • Assuring the QoS characteristics • Detection of untrusted entities	• Not guaranteeing the data control and ownership • Not assuring the process execution control • Not supporting the dynamic trust update • Complexity

- **Domain-based trust models**

  General description of this category

  Although, these models are applied in grid computing, there are some trust models proposed under this category for the cloud computing. In these models, cloud environments are divided into various autonomous domains and there are two types of relationships (i.e. within-domain and inter-domain) which are extracted from direct and recommended trust tables respectively. The values of within-domain trust rely on the transactions between the entities in the same domain. In case of needing to calculate the value of trust for another entity, the direct trust table will be checked; if the direct trust value (DTV) cannot be recognized, then the recommended value of trust from the other entities will be considered. The value of inter-domain trust is a proper value on the basis of the values from direct and recommended trust of the other domains [104, 105]. Collaborative trust models and security and interoperability centered trust models are the subtypes of this category.

  Collaborative trust models [95, 105] :

  In these models the cloud will be divided into several autonomous domains in such a way that each node keeps a trust table (TT) that maintains the value of trust related to all nodes which have been traded with that node in the domain. Each domain has a domain agent keeping three major tables including the domain inside trust table (DITT), the domain outside trust table (DOTT) as well as risk value table (RVT). Direct trust value (DTV) in TT will be increased or decreased based on the successful or failed transaction history with that node. In case of having no history of transaction in TT for certain nodes, the trust value in DITT (within domain) or DOTT (inter-domain) will be applied.

  - **Analysis of collaborative trust model [95]:** These models do not ensure the security requirements of users to establish trust on CSP, thus there is no guarantee to provide data integrity. Accordingly, there is not any functionality to enforce access control policies for

users; so they cannot ensure the data control and ownership provided by CSP. The detection of untrusted entities can be supported in these models by decreasing the trust value between two entities without any communication for a long time. However, there is not any support to process execution control and ensuring the QoS characteristics. Domain agent maintains the logs related to the all users' history records included failed and completed transactions with a specific CSP. These records are considered for updating the trust score. The algorithm for updating trust score and the time decay function has an easy applicability, thus has low computational complexity. Table 1-9 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-9. Strengths and weaknesses of collaborative trust models

Adapted from [95]

Strengths of Collaborative Trust Models	**Weaknesses of Collaborative Trust Models**
• Assuring the QoS characteristics • Supporting the dynamic trust update • Detection of untrusted entities • Low complexity	• Not assuring the data integrity • Not guaranteeing the data control and ownership • Not assuring the process execution control

Security and interoperability centered trust model [30, 95] :

Under this subtype of models, cloud users and cloud providers have user trust table and domain trust table respectively which contains the domain name, service type, trust degree and generation time. Each domain contains an agent of domain trust that is responsible for managing the trust tables to store the required characteristics for collaboration of cloud providers. In the first step when a user needs to assess the value of trust for a provider, it will match the needed domain name and type of service in the local user trust table. In case of having a greater trust value than the defined threshold, user will

start the transactions, otherwise the process will be suspended. In case of having no value in direct trust table, the recommended trust score will be applied.

- **Analysis of security and interoperability centered trust model [95] :** These models cannot simplify the assurance of data integrity and process execution control features in cloud environments. In these models, users can define a static threshold value for the required security along with recommendation factor of recommended trust applied in detection of untrusted entities. Moreover, these models do not consider the assurance of QoS characteristics which is one of the important features of trust models. Accordingly, there is not any method to ensure data control and ownership of users over the stored data. Domain agent is responsible for supporting dynamic trust update through time stamp of each transaction. Domain based trust model is easy to apply since within-domain trust evaluation depends on the number of entities in that domain, while inter-domain trust assessment is related to the number of domains. Hence, providing low complexity. Table 1-10 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-10. Strengths and weaknesses of security and interoperability centered trust models, Adapted from [95]

Strengths of Security and Interoperability Centered Trust Models	**Weaknesses of Security and Interoperability Centered Trust Models**
• Supporting the dynamic trust update • Detection of untrusted entities • Low complexity	• Not assuring the data integrity • Not guaranteeing the data control and ownership • Not assuring the process execution control • Not assuring the QoS characteristics

- **Subjective trust models**

  General description of this category

  These models divide trust into different subclasses such as authority trust [106], code trust [107] and execution trust. The main techniques to assess trust of a specified CSP are probability set theory and fuzzy set theory. These two approaches can be applied to assign weights and compute the individual trust subclasses. Then, a final value of trust by gathering these scores of trust can be computed. The novel weighted trust models and fuzzy comprehensive evaluation based models are the two subtypes of this category.

  Novel weighted trust models [95, 106] :

  The models in this category are considered as quantitative and qualitative cloud transform model which can express the randomness and fuzziness of trust values and their correlation. Cloud-based weighted trust model (CBWT) contains the weighted trust information transfer algorithm (WTIT algorithm) and weighted trust information combination algorithm (WTIC algorithm). CBWT model is based on the subjective trust nature which applies expectation (EX), entropy (En), hyper-entropy (He) to state an entity trust value. The WTIT algorithm is applied to transfer the trust information between two entities across certain path by applying recommended values of trust produced by other providers.

  - **Analysis of novel weighted trust models [95] :** These models do not ensure the security concerns of users such as data integrity and process execution control provided by CSPs. Detection of untrusted entities can be ensured by catering randomness and fuzziness in assessing trust. However, they do not support data control and ownership of users which is an essential feature for boosting the level of trust on CSP. Also, there is not any support to ensure the QoS characteristics that is fundamental to guarantee the delivery of services according to the users' specifications cited in the SLA. The applicability of cloud generation algorithm in reality introduces high complexity and partially supports dynamic

trust update and logging since there is no method to keep the history of transaction. On the other hand, it provides the dynamic trust update through CBWT model as the values of entropy and expectation can be updated constantly whenever the algorithm is performed. Table 1-11 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-11. Strengths and weaknesses of novel weighted trust models

Adapted from [95]

Strengths of Novel Weighted Trust Models	Weaknesses of Novel Weighted Trust Models
• Supporting the dynamic trust update • Detection of untrusted entities	• Not assuring the data integrity • Not guaranteeing the data control and ownership • Not assuring the process execution control • Not assuring the QoS characteristic • Complexity

Fuzzy comprehensive evaluation based trust models [95, 107] :

These models divide cloud trust system into two parts : service trust subsystem and user trust subsystem. A cloud information service center (CISC) is set up in the service trust subsystem that is the trusted third party and the trust management center for all the cloud providers. Trust assessment model presents the provider the three main characteristics that are provider identity, provider name and service trust. Trust evaluator performs trust fuzzy comprehensive evaluation (FCE) algorithm and assist to make a logical trust decision in choosing the most reliable cloud services.

- **Analysis of fuzzy comprehensive evaluation based trust models [95]**: These models do not encounter different essential security characteristics such as integrity, control and ownership of stored data on cloud environments. In addition, there is not any mechanism

to measure the QoS characteristics provided by CSP. Simple malicious nodes and role malicious nodes are the two malicious nodes but role malicious nodes are complicated to detect by applying FCE algorithm. Thus, these models provide mechanisms for partially detection of untrusted entities. Also, they do not ensure process execution control to the users. Deploying these models in reality and implementing the FCE algorithms have high complexity. These models support dynamic trust update and logging features since CISC retains the logs related to the transactions between CSPs and users and the algorithm for fuzzy trust evaluation provides a new judgment matrix of trust during each time of its execution to update the trust value dynamically. Table 1-12 summarizes the strengths and weaknesses of this subtype based on the defined assessment criteria.

Table 1-12. Strengths and weaknesses of fuzzy comprehensive evaluation based trust models, Adapted from [95]

Strengths of Fuzzy Comprehensive Evaluation Based Trust Models	**Weaknesses of Fuzzy Comprehensive Evaluation Based Trust Models**
• Supporting the dynamic trust update • Detection of untrusted entities	• Not assuring the data integrity • Not guaranteeing the data control and ownership • Not assuring the process execution control • Not assuring the QoS characteristic • Complexity

1.5.1 Discussion (Part 2)

Although, there may be more classifications of cloud trust models proposed in the literature, we tried to cover the maximum possibility of different classifications for trust models in section 1.4 of this chapter to provide a general view of their applicability. According to the aforementioned explanations, we summarize some significant findings as follows :

First, the analysis of the existing trust model classifications shows that classifying the existing cloud trust models in the literature is mainly based on different techniques which are used to assess trust and the criteria which are considered as the input of these models (such as the reputation and the feedback) in cloud environments.

Second, providing an accurate ranking in our analysis to clarify the best cloud trust model would be challenging and may not be accurate as each trust model in the literature has different applicability. Thus, ranking the cloud trust models absolutely depends on the users' requirements [95]. These requirements are represented as various characteristics which can be assessed by the trust model. Since there is not any proposed model to address various requirements, therefore, ranking a model as the best model would be meaningless.

Third, as users have diverse requirements, it is possible to combine various techniques in one unique trust model to calculate the score of trust. Therefore, in some cases a trust model can be categorized under more than one category (such as [108] that can be considered under subjective trust models category or under domain-based trust models category).

Fourth, as the notion of trust is associated with uncertain concepts, to some extent we can find the effects of various researchers opinion for classifying these models, as in the most classifications, the assessment criteria are not defined.

Likewise, as the trust model in this research project is supposed to evaluate a set of characteristics that are considered as complete as possible trust characteristics reference in the time of conducting this research based on assessing their standard measures, we propose « measure-based-behavioristic trust models » as a new category to accommodate our proposed model. The models under this category may combine different techniques to analyze the behavior of the provided cloud services based on the measures identified for the required characteristics. As the assessment criteria for this category, we propose the following criteria in addition to the seven points which are already explained and extracted from [95]. These

criteria have a noteworthy role in developing a comprehensive trust model. However, they are not addressed in the previous studies.

- Considering the role of related standards (i.e. quality and cloud computing standards);
- Assessing various aspects of trust in cloud computing to cover the maximum requirements (although QoS characteristics is one of the seven explained criteria in [95], here this point reflects different aspects of trust that QoS is one of them such as different aspects of objective and subjective trust related to the cloud services and also various key intrinsic characteristics of cloud computing which are defined in ISO/IEC 17788 standard);
- Defining standard measures for each identified trust characteristic and sub-characteristic;
- Applicability for several cloud users with different levels of access to the cloud resources.

Table 1-13 presents the potential drawbacks of the existing trust models under the discussed categories which will be addressed by the proposed solutions in the current research project. These solutions will be discussed in the next chapters of this thesis.

Table 1-13. The potential drawbacks of the existing trust models and the proposed solutions

#	The specified drawbacks of the trust models	The proposed solutions
1	Data integrity	Measuring Confidentiality and Authenticity as sub-characteristics of security.
2	Data control and ownership	Measuring Confidentiality and Authenticity as sub-characteristics of security.
3	Quality of service (QoS) characteristics	Measuring a set of characteristics that are considered as complete as possible trust characteristics reference along with their measures.
4	Detection of untrusted entities	Measuring required characteristics after collecting the feedback.
5	Dynamic trust update and logging	Comparing the behavior of cloud services in execution with the negotiated SLA and measuring Accountability as sub-characteristic of security by System log retention.
6	Process execution control	Measuring required characteristics based on the behavior of cloud services in execution.
7	Model complexity	Facilitate measuring the behavior of the services by : • Categorization of cloud users based on the level of access to the cloud resources. • Providing simple mathematical processes
8	Considering the role of related standards	Extracting the trust characteristics along with their sub-characteristics from quality and cloud standards : • ISO/IEC 25010 • ISO/IEC 25011 • ISO/IEC 25012 • ISO/IEC 19086-1 • ISO/IEC 17788
9	Assessing various aspects of trust	Evaluating subjective and objective trust, QoS characteristics and certain intrinsic features related to cloud computing.
10	Defining standard measures	Determining measures for trust characteristics based on : • ISO/IEC 25021 • ISO/IEC 25022 • ISO/IEC 25023
11	Applicability for cloud users	Categorizing cloud users based on the level of access to the cloud resources and proposing separate models related to the trust characteristics and sub-characteristics for each category.

In the next section, the common methodologies of existing trust models are reviewed.

1.6 Analyzing the methodologies of existing cloud trust models

The literature on evaluating trust in cloud environments has highlighted several techniques to assess trust characteristics. Therefore, this section explains the methods that are helpful for trustworthiness assessment of the cloud service providers as well as analysis of effective methods for cloud service provider ranking. These articles have been studied in context of their methods, techniques for trust evaluation, advantages and disadvantages. Moreover, these methods are selected based on the frequency of use and the efficacy of their performance in the cloud environments to deal with uncertain concepts.

Meanwhile the partial schemes of existing cloud trust models such as mathematical part of their methodologies give insight to the possible applicability of their features on the eventual model of this research.

1.6.1 Hypergraph based techniques

Hypergraph has emerged as a powerful graphical environment for analyzing and visualizing large amounts of data. Hypergraph is a competent technique to detect trends and patterns immediately. This enables the users to discover the complex relationships among multi-level entities. Scalability and managing huge amounts of data in a short time are the two dominant features of hypergraph.

Moreover, hypergraph is simply a generalization of the traditional graph [109] and a hyperedge can comprise one or more vertex. Therefore, hyperedge is as a result of grouping all these vertices. This feature of hypergraph is of interest since it is helpful to expurgate vertices with least importance or priority.

In terms of hypergraph techniques, many recent studies (e.g. [109, 110]) have shown the applicability of these techniques in evaluating trust in cloud environments. For example, Somu et al. in [111] presented a Hypergraph based Computational Model (HGCM) which is a cloud

service selection architecture for ranking cloud service providers. According to Somu [111], ranking models are significant contributory factors to the development of Trust Management Systems. Since there is a great variety of cloud service providers, prioritizing can be helpful for the cloud service provider selection.

In this work [111], Minimum Distance-Helly Property (MDHP) was applied as a hypergraph based technique. The role of MDHP is to find the relationship between Service Measure Index (SMI) characteristics according to the cloud users' requirements for choosing the proper cloud service provider. Also, MDHP can assign weights to the selected characteristics. Moreover, in order to impute missing values, Expectation-Maximization (EM) algorithms were employed.

Helly property is one of the substantial properties in the theory of hypergraphs [111]. According to this theory [112] :

> « A hypergraph has the Helly property if each intersecting family has a non-empty intersection (belonging to a star). It is obvious that if a hypergraph contains a triangle it has not the Helly property. A hypergraph having the Helly property will be called Helly hypergraph. »

On the other hand, Minimum Distance algorithm which is a supervised classifier learning algorithm, has a low-computational overhead and fast performance. It is suitable for classifying different types of N-dimensional signal. The main drawback of this algorithm is having a poor classification in case of choosing the wrong number of clusters [113].

The Expectation-Maximization (EM) algorithm can find maximum-likelihood estimations in a model which its data is not complete, has missing data or hidden variables. But EM has a too slow performance even on the fastest computers and cannot work properly when the big part of data is lost. The benefit of this approach is having good performance when the percentage of missing data is small and the data does not have too big dimensionality [114].

Table 1-14 summarizes the important drawbacks of hypergraph based techniques presented in [111].

Table 1-14. The drawbacks of hypergraph based techniques applied in [111]

#	The Hypergraph Based Techniques	The Potential Drawbacks
1	Minimum Distance-Helly Property (MDHP)	Having poor classification in case of choosing the wrong number of clusters [113].
2	Expectation-Maximization (EM) algorithms	Having good performance when the percentage of missing data is small and the data does not have too big dimensionality [114].

Based on the table 1-14, choosing the wrong number of clusters is unavoidable and the dimensionality of data may be huge as the requirements vary from user to user. Thus in the current research project, the simple distance formula is applied to avoid the explained problems (the details can be found in the next chapters).

1.6.2 Rough set theory

Rough set theory is a tool that mathematically can solve the problem of uncertainty, ambiguity and misty in data [115, 116]. Some of the significant advantages of applying rough set theory are its ability to solve the complex problems, addressing hidden patterns, data reduction, producing sets of decision rules and also can lead to interpret the final results easily [117]. Therefore, rough set theory has been catching researchers' attention recently especially in evaluating cloud services. Some good examples for applying rough set theory in proposition of cloud trust model are [42, 48, 110].

Beside all the advantages of the rough set theory, there are some drawbacks that need to investigate more. One of these drawbacks is dependency of rough set on complete data systems [118]. Having complete data systems in real life is unrealistic due to access restrictions and errors which are occurred in measurements and dis-operation [118].

In a study conducted by Li et al. [42], a cloud trust model was proposed in which a novel combination of rough set and Induced Ordered Weighted Averaging (IOWA) operator is applied to trust data mining and knowledge discovery. Rough set is used for knowledge discovering from trust characteristics and IOWA operator is applied to merge the total trust values based on time series to have better real-time performance.

Considering the dependency of rough set on complete data systems (and the fact that having complete data systems in real life is unrealistic due to access restrictions) [118], knowledge discovering from trust characteristics in this work [42] may be challenging as the levels of access to the cloud resources are not clarified. This fact highlights the importance of categorization of cloud users based on their level of access in cloud environments for applying cloud trust models which is addressed in the current study.

1.6.2.1 Induced Ordered Weighted Averaging (IOWA)

IOWA operators enable reordering the arguments by using an additional induce variable [119, 120]. IOWA is an extension of the OWA operator which is a common aggregation method [121]. Aggregation operators cause to create a single value from a set of values and makes an abstraction of inputs [122].

A key aspect of the IOWA operator in comparison with the OWA operator is its capability to deal with complicated reordering process when the first value is not the highest one in the reordering [123]. However, as explained in [42, 124] the significant issue in IOWA theory is defining the related weights, as the calculated weights may not be accurate since the prioritization of the values by the users are not considered. This issue is addressed in the current research project by applying Full Consistency Method (FUCOM) and will be discussed in the next chapters.

1.6.3 Fuzzy logic theory

Having rigid mathematical framework, fuzzy theory addresses the vague and the indecisive concepts in a precise and rigorous way [125]. In addition, it can be used as a modeling language in case of existing fuzzy relationships [125]. Accordingly, taking into account the notion of trust, these concepts lend support to the idea of utilizing fuzzy logic theory by several researchers as a method for evaluating trust in cloud environments. As an example, in [126] the authors developed a fuzzy logic-based model that is able to receive the user's feedback in the format of fuzzy linguistic terms. In addition, the proper weight of this feedback is calculated by applying fuzzy inference system. Finally, fuzzy goal is incorporated to determine the trust value regarding the feedback weight.

In this context it may be beneficial to mention the main goals of fuzzy theory succinctly. Zimmermann in [125], labelled and explained these goals as following:

- Modeling of uncertainty: this is the most important goal of fuzzy theory. Uncertainty can be modeled based on the causes of uncertainty, the type of information, the requirements, etc.;
- Relaxation: fuzzy theory is capable to relax or generalize traditional methods from dual logic (e.g. feasible and infeasible, belonging or not etc.);
- Compactification: fuzzy theory causes to reduce the complexity of data, usually via linguistic variables or fuzzy data analysis.

Thus far, various studies have utilized fuzzy theory for trust assessment in cloud environments (e.g. [47, 50, 127-130]). However, this theory has some disadvantages that may impact on the final result of evaluation [131] :

- Generating fuzzy rules and membership functions may be tedious;
- Fuzzy outputs can also be interpreted in different manner which makes the analysis laborious.

Subsequently, in spite of the fact that fuzzy theory is able to remove the problems related to the indecisive concepts, based on the aforementioned drawbacks and as trust in cloud computing has no agreed-upon rule to be evaluated, it seems that *evaluating* a vague concept in a dynamic environment such as cloud by applying a method such as fuzzy logic theory that might lead to produce different interpretations of results, would not be reliable. The main reason is that the outcome of trust assessments must be decisive to be helpful in making decision of cloud service provider selection as well as to be effective in raising users' confidence. If the outcomes are interpreted in different manners, the efficacy of evaluation will be diminished. This fact emphasizes on the importance of the selected methodology for trust evaluation that needs to be potent for producing reliable results.

1.6.4 Markov Chain

Markov chain is known as a mathematical system that represents transitions between the states under certain probabilistic rules. It is a simple way to model random processes statistically. 'Finite state machines' and 'random walks' can model the Markov chain [132].

A number of studies have begun to examine the applicability of this method in proposing cloud trust model. For example, Chandrasekar et al. [133], introduced a dynamic trust model based on Markov chain. In this work, trust was calculated dynamically according to the way that services were provided by cloud providers. Moreover, as it is discussed in this article:

> « This model can be extended into an Absorption Markov Chain with a new state introduced called Ideal State, which will be the absorption state. An absorption state is a state in the Markov chain which when reached, it is not possible to move to any other state from it. There is no 'out transition' from an absorption state. A Markov chain with such a state is known as Absorption Markov chain. »

As other benefits of Markov chain, its simplicity to apply and understanding as well as conducting complicated calculations easily, can be considered.

On the other hand, one of the main disadvantages of Markov chain is that the computation amount will increase promptly with the number of states and time [134]. As one of the assessment criteria of the cloud trust models is the model complexity, this drawback of the Markov chain can increase the complexity of the model and make it error-prone. Therefore, to develop a model with low complexity, Markov chain may not be a proper method.

1.6.5 Pearson Correlation Coefficient (PCC)

Correlation among variables means measure of their relationship. A well-known measurement of correlation is the Pearson correlation coefficient. Pearson Correlation Coefficient is used to measure the stability of relevance between two variables. It depicts the linear relationship between data. It is believed that Pearson Correlation Coefficient is the best method for measuring the association between interest variables as it is based on the covariance method [135].

Accordingly, in a study which set out to propose a trust service-oriented scheduling model for workflow applications in cloud computing, PCC was applied to calculate weights for the cloud users [136]. Although, PCC is a widely used method to calculate the similarity of the relationship between two variables [136], it is not able to distinguish the difference between dependent and independent variables. Therefore, it is necessary to be aware of data which is used by PCC [114].

In this research project to overcome the mentioned issue of applying PCC in weights calculation of trust characteristics, the Full Consistency Method (FUCOM) [137] will be considered. As explained in [137], one of the main advantages of FUCOM in comparison with the existing multi-criteria decision-making (MCDM) methods is calculating reliable values of

criteria weight coefficients related to the rational judgment (the details related to this method will be found in the next chapters).

1.6.6 The Delphi method

The aim of this method which is a structured communication method is to elicit the expert opinion. According to the explanation of Delphi method in [138] :

> « Usually, Delphi is conducted through a series of questionnaires. The panel members remain unknown to each other (Martino, 1983; Robinson, 1991). Following each round, the responses are analyzed, and based on the analysis, a new questionnaire is developed and sent to the panel members in the next round… »

This method has three characteristics [138] :

- Anonymity: as mentioned above, interaction under Delphi method is completely in an anonymous way. This can help to examine any opinion with minimum pressure on the group;
- Repetition with Controlled Feedback: as Delphi is carried out by the questionnaires, in each round, the participants will be provided with feedback which contains new information. Consequently, it causes to reduce the willing of participants to reach a consensus for generating an effective opinion;
- Statistical group response: it represents the overall opinion of the group.

Additionally, this method has various advantages such as being flexible for accommodating many variations and applications [139]. Therefore, numerous studies have attempted to use Delphi method in their proposed cloud trust models (such as [62, 140]). As an example for the applicability of this method in trust evaluation, in [62] a trust label system is developed through the Delphi methodology. By applying this methodology, the trust label interface is extracted. This interface specifies a range of essential metrics such as data security, service

levels and backup of data that are necessary in the Delphi participants opinions for communicating trust in cloud services. However, there are paucities when it comes about the use of this method. For example [139] :

- The method requires an effective guidance and agreed standards to interpretation of the results;
- This method is not efficient as a result of producing new knowledge;
- Being anonymous may cause to less ownership of the ideas.

To overcome the aforementioned problems in this research project, we consider related ISO/IEC standards to derive the important characteristics as well as the characteristics that were proposed in domain-related research papers. Subsequently, we identify a set of trust characteristics that will cover different aspects of trust to be evaluated for the offered cloud services (more details can be found in the next chapters).

1.6.7 Analytic Hierarchy Process (AHP)

AHP is one of the most practical ways of Multi Criteria Decision Making (MCDM) method to extract ratio scales from paired comparisons [141]. As discussed in AHP tutorial [141] :

> « The input can be obtained from actual measurement such as price, weight, etc., or from subjective opinion such as satisfaction feelings and preference. AHP allows some small inconsistency in judgment because human is not always consistent. »

Since selection of a reliable cloud service provider is a Decision Making problem, so the most challenging task in this problem is choosing the main factors that have an important role for that decision [142]. AHP is capable to organise these factors hierarchically from the main goal to criteria, sub-criteria and alternatives levels [142]. Arranging factors hierarchically is helpful for providing an overall view of the complicated relationships [142].

Accordingly, numerous researchers introduced different applications of AHP method in selecting a proper cloud service provider. As an example, Alhanahnah et al. in [60], by the help of AHP, enabled customers to determine the priority of their preferences and improved the proposed Trust Evaluator to support various ranges of trust indicators.

The major advantages of AHP are; facilitating solving complex decision problems by decomposition of a decision problem into subparts, its flexibility and having a powerful method to check the consistency and also help to obtain both subjective and objective measure of the evaluation [143].

Despite the popularity of this method, there are certain drawbacks associated with the use of AHP. The most significant disadvantages of AHP as discussed in [144] is that : since AHP methodology divides the decision problem into various sub parts, can cause to create a lengthy task. Moreover, it may be difficult to figure out the suitable scale for each selected preference.

Regarding the disadvantages of AHP, we apply Full Consistency Method (FUCOM) to determine the priority of the cloud user's preferences. More detail on this method along with its advantages can be found in next chapters.

1.6.8 Discussion (Part 3)

The analysis of used methodologies in cloud environments for helping to evaluate trust undertaken here, has extended our knowledge of various mathematical useful methodologies in this area as well as their drawbacks. However, still there are a wide variety of methodologies proposed in the literature that need greater efforts to identify.

Thus far, the conducted literature review in this section, has highlighted some empirical findings :

- Firstly, it is important to bear in mind the incorporation of uncertainty in trust evaluation [60]. Because the notion of trust is combined by indeterminacy and ambiguity. So it may not be accurate to evaluate this vague concept by methods and techniques which may not produce decisive outcomes or their results might be interpreted in different ways (such as fuzzy logic theory or Delphi method);

- Secondly, the literature review revealed that a considerable part of trust evaluation in cloud environments associates with data analysis techniques. Because, in order to determine the overall trust, data must be gathered from various resources [60]. Therefore, in order to have an accurate trust evaluation, the credibility of the gathered data before taking them into account in the evaluation, should be ensured. Regarding the features of three cloud service models (Infrastrusture as a Service, Platform as a Service, Software as a Service) which will be discussed in the related chapter in this research project, cloud service users may have restricted access to various cloud resources in cloud environments. Accordingly, from cloud service users' perspective, collecting data from cloud resources to evaluate trust may be challenging and in some cases might be impossible. Given that various cloud trust models are designed, to the best of our knowledge there is no substitute solution for this problem in the literature. This fact puts emphasis on the necessity of cloud users categorization in terms of applying trust models to evaluate trust in cloud computing. Therefore, in this research project we categorize cloud service users based on their level of access to the cloud resources to solve this issue (more details are explained in the next chapters);

- Thirdly, it is worthy to note that selection of a trustworthy cloud service provider is a kind of decision-making process. Accordingly, there are numerous methods proposed by different scientific researchers for solving the problems in decision-making process in various fields. These methods are broadly applied in creation of trust models for trust evaluation in cloud environments. However, the determined methods in the existing models have certain drawbacks that may impact on the final results of the trust assessment. As an example we refer to AHP. Although, AHP is one of the popular methods in MCDM, this method by creation of lengthy tasks may cause to increase the complexity of the model and

consequently, make the model error-prone. This fact clearly highlights the importance of selection of a precise method in trust evaluation;

- Finally, this study has provided a deeper insight into trust methodologies in cloud environments by analyzing their advantages and drawbacks. Despite the fact that each identified methodology in the literature may has some disadvantages, further research revealed that it is important to determine the best methodology with less implication on the result of trust evaluation in cloud computing. For example, certain methods are not able to distinguish the difference between dependent and independent variables (such as PCC) or do not consider the prioritization of the values by users (such as IOWA) which is one of the main features in assessing trust in cloud environments as each user has different requirements. However, these methods are known as the best method for measuring the association between interest variables and having suitable process to deal with complicated reordering process, respectively. Therefore, the current research seeks to explain the development of a model to represent the behavior of the cloud services according to the context of usage and at the same time by having potent framework, would be able to evaluate trust. As will be explained in the next chapters, to overcome these problems Full Consistency Method (FUCOM) can be a proper substitute.

1.7 Chapter summary

The literature review presented in this chapter is divided into five main sections :

In the first section, we investigated the main aspects of trust that are considered as trust characteristics such as security, reliability, performance efficiency and so on to respond to the question of « what » should be evaluated in the process of trust assessment in cloud computing. In other words, « what » are the main characteristics on which are focused in the literature.

In the second section, we explained the applied approach in exploring state-of-the-art on trust models to clarify the necessity of this investigation and put emphasis on the fact that each of

the existing model has a defined purpose. For example some of them are developed to evaluate security (e.g. [97]), the others for assessing certain characteristics of QoS (e.g. [11]) or identifying fake feedback (e.g. [145]) and so on. As explained, this may be one of the main reasons for the lack of a multipurpose model to evaluate different aspects of trust with various applicability.

In the third section, we discussed the common features of trust models to provide a general overview. This can be considered as the basis to help us in better understanding of these models.

In the fourth section, we clarified the concept of trust and trust models in cloud computing and elaborated on various classifications provided in the literature. Furthermore, we described the weaknesses of the models under each classification and proposed certain substitute solutions. In addition, we identified a new category of models to accommodate our proposed model based on its unique features.

Finally, in the fifth section we analyzed the commonly used methods in the trust models and identified their main weaknesses in trust assessment. For these weaknesses we proposed some solutions as well that will be addressed in the proposed model of current research.

In summary, it seems that all the weaknesses that were identified by investigating the literature can be considered as the guidelines for developing a model in which the main drawbacks of the existing models will be addressed (table 1-13 and section 1.5.8). Thus, the main purposes of the presented literature review were :

- Identifying the main problems related to the trust;
- Specifying the important and common features of trust models;
- Determining the major weaknesses of the existing models related to their classifications and methods for trust evaluation.

The second chapter is concerned with the methodology used for conducting this study. In this chapter the methodological aspects of the research, research objectives and the main research steps used to design ED-BeCT are discussed.

CHAPTER 2

RESEARCH METHODOLOGY AND OBJECTIVES

This chapter is concerned with the methodology used for this research project. Section 1 describes research issues and restates the fundamental research questions. Section 2 presents the principal goal and objectives of the research project. Section 3 elaborates on the details of the major phases to accomplish the objectives of the research. Finally, section 4 concludes the chapter.

2.1 Research project description

Regarding the notion of trust, the idea of evaluating this concept in cloud environments may seem challenging at the first glance. However, the fact that there are some organisations or even some experts to evaluate the trustworthiness of the cloud service providers, gives motivation to develop a model to address the critical requirements, in particular for helping cloud customers to select a proper cloud service provider.

The research issues which are identified by conducting and analyzing the literature (chapter 1) refer to constraints in identifications of cloud service users' requirements regarding the features of three cloud service models (i.e. Infrastructure as a Service - IaaS, Platform as a Service - PaaS, Software as a Service - SaaS). In addition, since there is no explanation of trust in the related cloud computing standards such as ISO/IEC 17788, the existing cloud trust models suffer from lack of standardization [54].

On the other hand, the Information Technology Infrastructure Library (ITIL) which is a set of practices and policies to manage Information Technology (IT) infrastructure, development and operations [146], can be applied in the migration process of businesses to cloud environments [147]. However, as discussed in [147] the serious deficiency of the published works is that there is not a systematic process with adequate details to be considered as a guide for the users

throughout the steps and decisions related to the migration of data to cloud and selection of a proper cloud service in conformity with the identified requirements.

Subsequently, in this research project we aim to overcome the aforementioned constraints and the limitations which are discussed in chapter 1, by developing a trust model of which the trust characteristics are mainly the intrinsic characteristics of cloud computing explained in cloud computing standards as well as the quality characteristics that are extracted from system and software quality standards.

The method of this trust model is supposed to be convenient in usage and be able to meet the maximum requirements of cloud service users. In addition, this would be able to assess the measures of quality characteristics that are supported by ISO/IEC 25022 and ISO/IEC 25023 to infer the value of trust as accurate as possible.

Moreover, the model in this research project will apply the dedicated process to assess the provided cloud services. This process which is depicted in figure 2-1, will be helpful for the users who wish to benefit from three cloud service models (i.e. IaaS, SaaS, PaaS) to specify the ideal values for the characteristics that are mentioned in the negotiated SLAs with cloud providers, filter the qualified cloud service providers (CSPs), evaluate the selected quality and trust characteristics and finally, make a decision to select (a) proper cloud service provider(s).

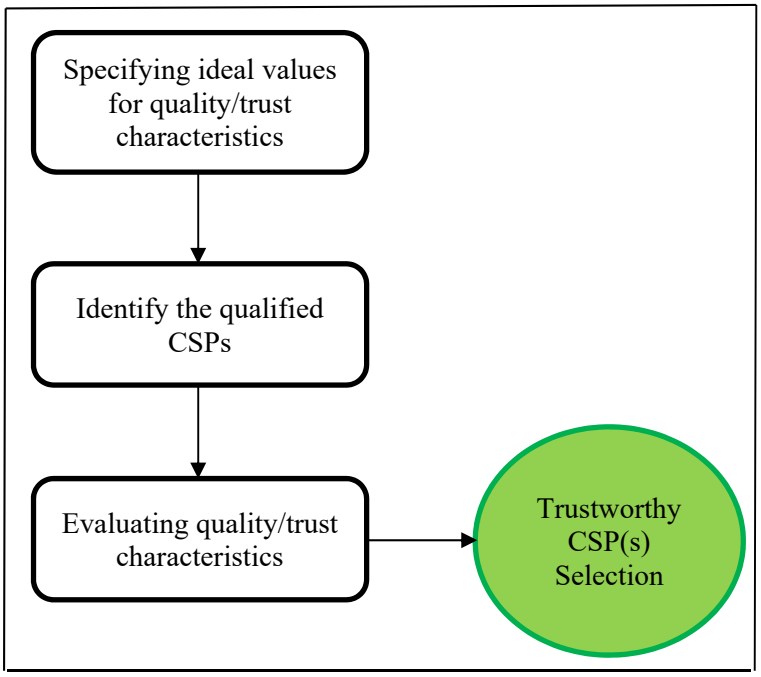

Figure 2-1. The overall process in the proposed model

Accordingly, the basic questions related to the process depicted in figure 2-1 in this research project are organized around four essential elements :

- Derive the maximum set of quality and trust characteristics which can be as complete as possible in the time of conducting this research based on the related ISO/IEC standards;
- Derive the proper measures for the selected quality and trust characteristics according to the related ISO/IEC standards;
- Identify the qualified cloud service provider(s);
- Aggregate the results of measuring these characteristics to calculate the level of trust as accurate as possible.

2.2 Research Goal and Objectives

This work builds on the knowledge to develop a broadly acceptable cloud trust model (**E**nhanced **D**ynamic - **Be**havioral **C**loud **T**rust model (ED-BeCT)) to evaluate trust based on

standard measures in the cloud environments. Hence, in order to achieve this goal the following objectives are specified :

- Identifying the key trust characteristics that could be recognized in the system and software quality standards and cloud standards by analyzing:

  - trust characteristics of the current cloud trust models in the literature;
  - system and software quality standards;
  - cloud standards.

- Analyzing and specifying the weaknesses of the current cloud trust models by:

  - Identifying the models that they have certain consensus published in the renowned journals and software quality publications;
  - Comparing the identified models to see what they have in common and what are their empty spaces by matching them with cloud standards (ISO/IEC 17788).

2.3 Research Methodology

The purpose of this section is to present the research methodology. It should be noted that the research methodology in this work is divided into five major phases which are discussed as follows :

- **Analysis phase** : This phase is applied to study the main concepts in and related definitions to the domain of trust in cloud computing. This phase includes a literature review of :

  - The main quality and trust characteristics (details in chapter 1);
  - The proposed measures of these characteristics (details in chapter 1);
  - The existing trust models classifications (details in chapter 1);

- The methods and techniques to assess the trustworthiness of cloud service providers (details in chapter 1).

The literature review which has been conducted in this phase of the research, led to the following results (more details in chapter 1):

- There is no consensus on quality and trust characteristics;
- There is no standard measure for the trust characteristics;
- There are various classifications of trust models which makes the analysis of these models difficult as each model has different applicability;
- There is no consensus on any of the existing cloud trust models to be broadly acceptable.

The proposed trust models in the literature partially or completely ignore the effects of existing system and software quality standards in the final results. Thus, the calculated or induced trust may vary by using various existing trust models. Subsequently, to the best of our knowledge there is no broadly acceptable cloud trust model in the literature.

- **Matching phase** : This phase of the research includes matching the proposed characteristics specified by investigation in and analysis of the literature, with the characteristics and sub-characteristics which are discussed in related ISO/IEC standards. Further, we match the potentially applicable measures extracted from related standards with these characteristics. Later, these characteristics and sub-characteristics along with their measures will be evaluated by the final trust model of the current research project.

  The reason of matching phase is that the main goal of this research project is to develop a broadly acceptable cloud trust model which would be able to cover as complete as possible set of trust characteristics in the time of conducting this research. Thus, in this way this model could address the maximum requirements that are identified so far. Furthermore, as these trust characteristics are mainly extracted from related standards, the results of

assessing these characteristics would be more reliable since they are assessed based on the standard sub-characteristics along with their standard measures.

The matching phase is divided into two stages (details in chapter 3 and chapter 4):

The first stage : Matching the specified characteristics in the literature with the characteristics and their sub-characteristics in related standards : the purpose of this stage is to identify the quality characteristics and their sub-characteristics (regarding various aspects of trust which are analyzed in previous phase) which are explained in ISO/IEC 25010, ISO/IEC 25011, ISO/IEC 25012, and match them with the specified characteristics (based on various users' requirements and their expectations from cloud services discussed in chapter 3) in the literature. In addition, the intrinsic characteristics of cloud computing extracted from ISO/IEC 17788 are matched with the characteristics that are supposed to be evaluated as trust characteristics (further details in chapter 3).

The second stage : Matching the measures with the identified characteristics and sub-characteristics : the purpose of this stage is : 1- to extract related measures for the characteristics and their sub-characteristics from ISO/IEC 25022 and ISO/IEC 25023, 2- to clarify the internal relationships of the trust characteristics to be able to specify the related measures for them as for the characteristics defined in cloud standards there is no identified measures. In addition, certain intrinsic characteristics of cloud computing among the proposed characteristics have not any defined measures with other characteristics, therefore, we propose suitable measures for them based on the related discussions in chapter 3 and chapter 4 (further details in chapter 4).

- **Tracing phase** : this phase includes an identification of IaaS, PaaS and SaaS to find their particular features, similarities and differences to verify the applicability of the proposed measures for the trust characteristics and their sub-characteristics (chapter 5). Accordingly, the feasibility of measurements for the three cloud service models will be discovered in this phase. Finally, based on the results of this phase three different models of trust

characteristics for the three different types of cloud users that are discussed completely in chapter 6, will be presented. Thus, the tracing phase can be divided into two stages :

The first stage : An investigation into IaaS, PaaS, SaaS : the purpose of this phase is to specify the main features of IaaS, PaaS and SaaS to be able to determine the applicability of proposed measures in cloud environments. In addition, as explained in chapter 6, cloud users will have different levels of access to the cloud resources in case of using IaaS, or PaaS or SaaS. This fact can influence on assessing the provided cloud services (further details in chapter 5).

The second stage : Verification of the applicability of the identified measures : the goal of this stage is to clarify the dependency of the identified measures to the cloud users' level of access to the cloud resources. As some of the measures rely on having direct access to the cloud resources to be evaluated, therefore, we separate the models of the trust characteristics for each category of cloud users (further details in chapter 6).

It should be noted that the results of tracing phase are presented in chapter 7 completely.

- **Development of ED-BeCT**: this phase includes the design of a cloud trust model and the related procedures addressing the identified requirements of the cloud users. This phase is divided into two stages :

The first stage : Design of the filtration stage : this stage elaborates on the process of filtering the qualified cloud service providers. As cloud computing is an evolving technology, the number of cloud service providers is projected to grow significantly each year. Therefore, the goal of this stage is to facilitate the process of cloud service provider selection and ensure that the cloud service assessments will be performed among competent providers (further details in chapter 8).

The second stage : Design of the assessment process in ED-BeCT : this stage describes the main assessment processes of ED-BeCT in detail. In addition, this stage includes all the necessary measurement functions, equations, and related procedures to calculate the level of trust related to the cloud service provider (details in chapter 8).

- **Exploitation phase :** In this phase an example is designed to certify the applicability of the proposed model and evaluate to what extent this model would be able to cover the cloud users' requirements. Further, the different steps of this model are clearly explained to provide a better understanding (chapter 9).

Figure 2-2 depicts the different phases of the methodology of this research project.

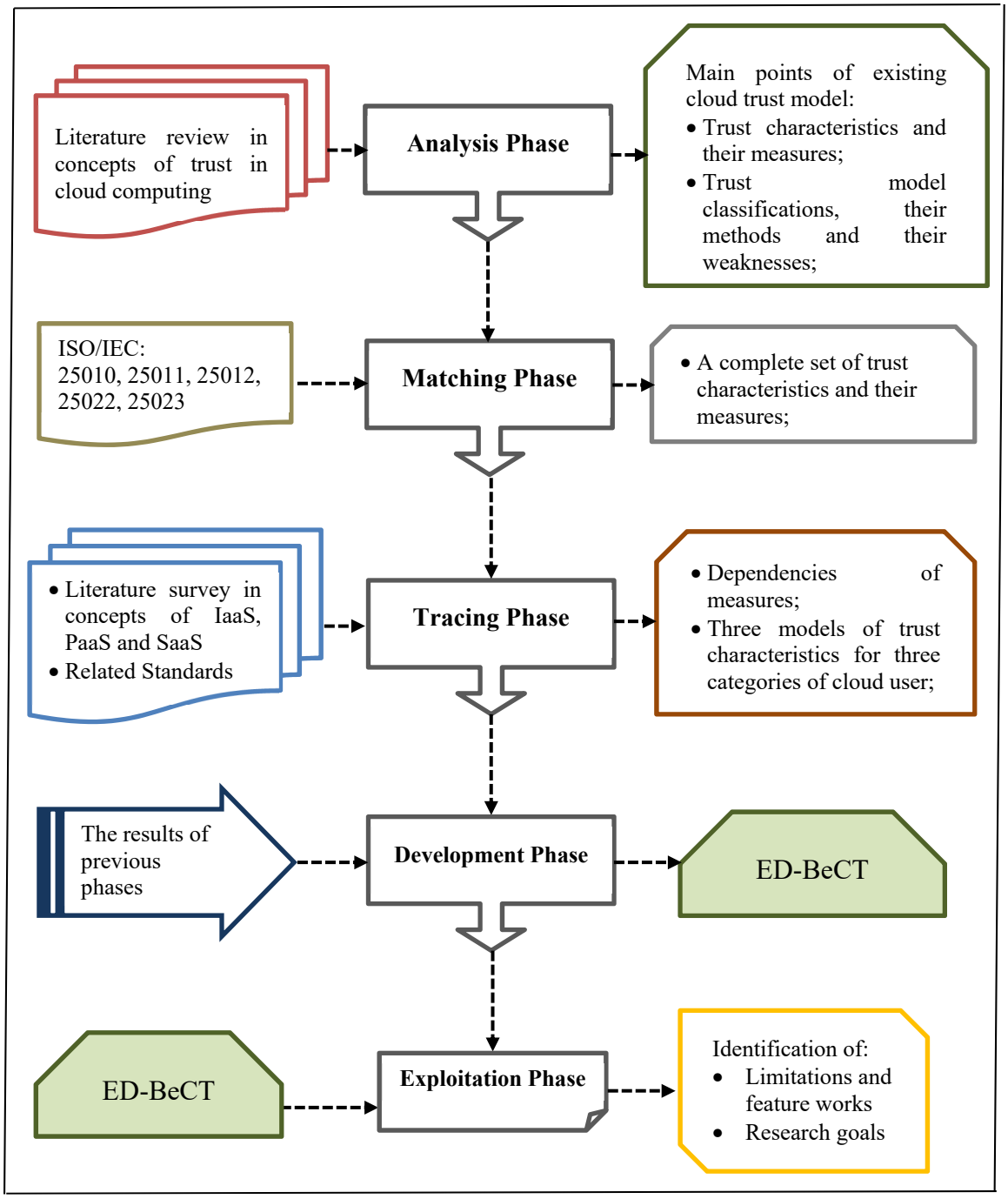

Figure 2-2. The phases of the research methodology

2.4 Chapter summary

This chapter describes the research methodology, restates the fundamental questions and the objectives of the research project to develop ED-BeCT. Figure 2-2 in this chapter, illustrates the path through which we formulate the problems, specify the gaps in the literature on the cloud trust models and fill these gaps (table 1-13 and section 1.5.8 in chapter 1) through various dedicated procedures.

The research path to achieve the stated objectives has been explained through a dedicated five-phase methodology which are : analysis, matching, tracing, development and exploitation. As described, in the analysis phase a comprehensive literature review is conducted to identify main characteristics which are proposed as trust characteristics in the literature. Further, an investigation among various scientific and well-known journals and conferences is carried out to analyze the insufficiency of previous studies in the domain of characteristics measures, the cloud trust models classifications and the proposed methods and techniques to evaluate the trust characteristics. The matching phase is related to match the identified characteristics from the literature with the related standards to develop a more comprehensive set of trust characteristics with their sub-characteristics and their standard measures. The tracing phase provides a detailed overview of the main features of IaaS, PaaS and SaaS to verify the applicability of the proposed measures in cloud environments. The development phase explains the process of ED-BeCT for filtering qualified cloud service providers and evaluating the provided cloud services. In addition, in this phase different stages are described to calculate the level of trust related to the filtered cloud service providers. Finally, the exploitation phase shows the stages of the ED-BeCT through an example.

The next chapter will elaborate on the first stage of matching phase.

CHAPTER 3

THE PROPOSED SET OF TRUST CHARACTERISTICS IN CLOUD ENVIRONMENTS

This chapter describes the applicable ISO/IEC standards in cloud computing and matches the identified characteristics in the literature with their standard characteristics and sub-characteristics. Accordingly, there are two main goals in this chapter: first, to verify the applicability of these standards in cloud computing. Second, to identify the basic set of trust characteristics based on the analyzed ISO/IEC standards.

Accordingly, section 1 gives a brief overview of the applicable standards and describes their different parts that can be related to the essential characteristics and features of cloud services. Section 2 is dedicated to matching trust characteristics extracted from the literature with the explanations in the standards. Section 3 presents the results of this matching and finally, section 4 concludes the chapter.

3.1 ISO/IEC Standards Applicable in Cloud Computing Technology

In this section, the quality and cloud characteristics based on the existing ISO/IEC standards are identified. These characteristics are defined in international standards and the aim of this identification is to derive the basic set of characteristics to be evaluated by the proposed cloud trust model (ED-BeCT).

3.1.1 ISO/IEC 25010

Trust and quality have a direct relationship; therefore, a comprehensive understanding of quality models can help to develop a cloud trust model with respect to the required trust characteristics. The definition of the quality in ISO/IEC 25010 [1] states: “[quality] is the degree [to which] the system satisfies the stated and implied needs of its various stakeholders, and thus provide values”. To address the meaning of this definition three quality models were

developed: Quality in Use (QiU) and Product Quality (PQ) models in ISO/IEC 25010 and Data Quality model in ISO/IEC 25012. In the QiU model, trust is a sub-characteristic of Satisfaction characteristic (figure 3-1).

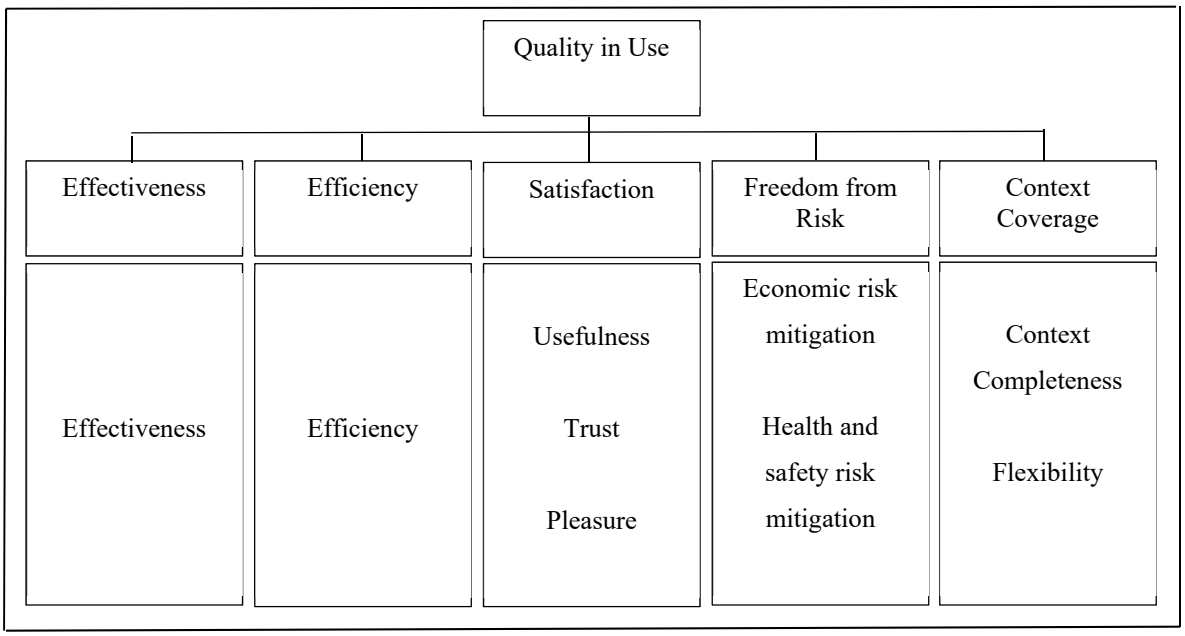

Figure 3-1. Quality in Use model
Taken from ISO/IEC 25010 [1]

As the literature revealed in chapter 1 and it is depicted in figure 1-1 in this chapter, security is a prominent trust characteristic, which is one of the characteristics in product quality model. Like other characteristics in this model, security has some sub-characteristics as well. It is recommended that security be considered as presented in ISO/IEC 25010 (figure 3-2) with its sub-characteristics in existing or being developed trust models.

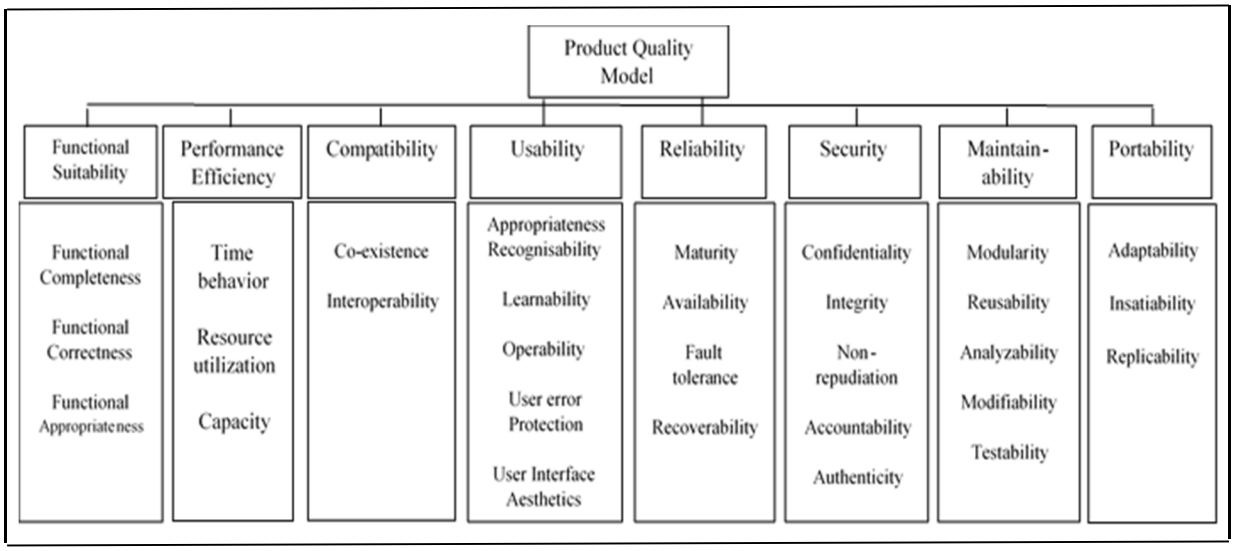

Figure 3-2. Product quality model
Taken from ISO/IEC 25010 [1]

Trust in ISO/IEC 25010 [1] is the "degree to which a user or other stakeholder has confidence that a product or system will behave as intended". It is almost certain that considering ISO/IEC 25010 characteristics can be helpful in designing the cloud service trust model that would satisfy most of the user's needs.

3.1.2 ISO/IEC TS 25011

ISO/IEC TS 25011 defines a model for quality of the IT services. The defined components of IT services are people, processes, technology, facilities and information [148], and as such make a considerable part of every cloud service. In consequence, the characteristics of the ISO/IEC 25011 would be recommended as the part of each cloud service trust model. Fig 3-3 presents the IT service quality characteristics published in ISO/IEC TS 25011.

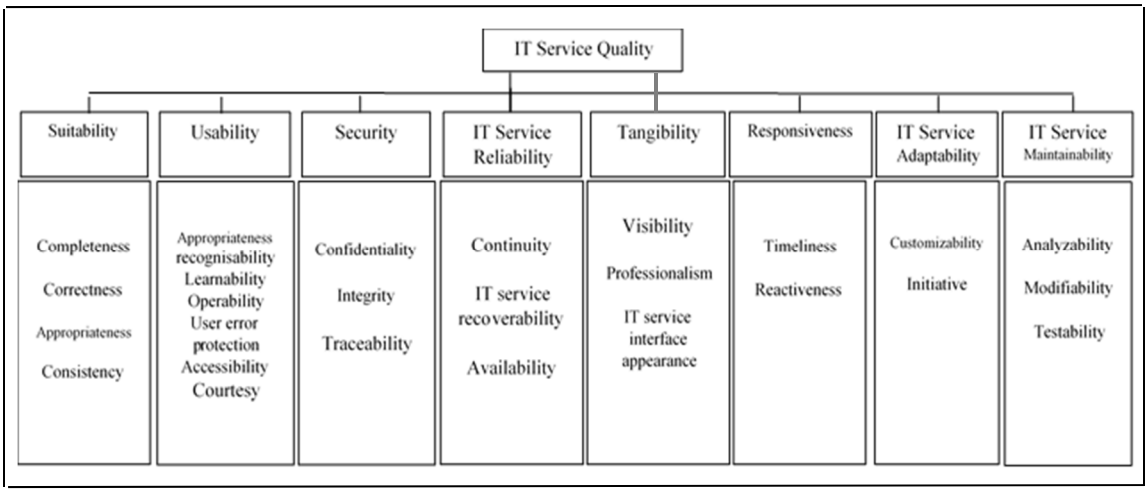

Figure 3-3. IT service quality model
Taken from ISO/IEC 25011 [148]

3.1.3 ISO/IEC 25012

Another important factor related to cloud services' trust is the retention of data quality while being processed by the service. The data is the principal component of each service/communication and it is believed that the precision of the results depend on its correctness [149]. To address this subject ISO/IEC 25012 presents data quality model with the recommended characteristics.

Taking into consideration the importance of data quality, particularly when data is being processed in cloud environment it is recommended that eventual cloud services trust model take into consideration approach published in ISO/IEC 25012.

Table 3-1. Data quality characteristics.
Adapted from ISO/IEC 25012 [149]

Data Quality Characteristics	
Accuracy	Efficiency
Completeness	Precision
Consistency	Traceability
Credibility	Understandability
Currentness	Availability
Accessibility	Portability
Compliance	Recoverability
Confidentiality	

3.1.4 ISO/IEC 17788

In order to design a comprehensive trust model in cloud environment, the key characteristics of cloud computing should be defined. ISO/IEC 17788 defines these key characteristics and cross-cutting aspects of cloud computing which are illustrated in the table 3-2 [24].

It is necessary to emphasize that such aspects (defined in ISO/IEC 17788) may impact multiple roles, activities, and components, in such a way that it is not possible to clearly assign them to individual roles or components, and thus become shared issues across the roles, activities and components.

As it can be seen in the table 3-2 several of the cross-cutting aspects defined in ISO/IEC 17788 are presented in form of characteristics (and sub-characteristics) in ISO/IEC 25010. This already existing link suggests that the eventual model for cloud services trust should take into consideration the superposition of both concepts.

Table 3-2. Cloud computing characteristics
Adapted from ISO/IEC 17788 [24]

Key Characteristics of Cloud Computing	**Cross-Cutting Aspects of Cloud Computing**
• Broad network access • Measured service • Multi-tenancy • On-demand self-service • Rapid elasticity and scalability • Resource pooling	• Auditability • Availability • Governance • Interoperability • Maintenance and versioning • Performance • Portability • Protection of personally identifiable information (PII) • Regulatory • Resiliency • Reversibility • Security • Service level and service level agreement (SL and SLA)

3.1.5 ISO/IEC 19086-1 (FDIS)

As it is illustrated in the table 3-2 and according to ISO/IEC 17788, SLA is one of the cross-cutting aspects of cloud computing. Therefore, it is necessary to incorporate key characteristics of cloud computing in the cloud SLA to simplify the collaboration between cloud service provider and cloud service user [150]. Hence, the first part of ISO/IEC 19086 clarifies principal concepts and definitions for the cloud service level agreement (SLA).

3.1.6 ISO/IEC DIS 19086-3

In the third part of the ISO/IEC 19086 the core requirements for cloud service level agreement are specified. It also clarifies the cloud SLA content areas and discusses their components.

This standard is useful for both cloud service users and cloud service providers in order to adjust the agreement. By developing an accurate SLA (between cloud service user and cloud service provider) that observes related standards, an important part of creating trust would be established.

3.2 Comparing trust characteristics: research versus standards

Choices of trust definitions and characteristics are usually based on the context of use. However, in the standards that are related to cloud computing not only there is no consensus-based definition for trust, but there is also no sufficient number of consensus-recognized characteristics that could potentially constitute the trust model.

The literature sources discuss several various ways of evaluating the trust in the context of cloud services. In some cases, trust can be considered "gained" when there is enough evidence to prove the system meets a set of pre-defined requirements [19]. Several proposed cloud trust models employ these evidences as trust characteristics [47]. In addition, some cloud trust models are based on the reputation of the cloud service provider [151]. It is important to recall here that trust and reputation may have related concepts, but often they have different meanings [12] (details in chapter 1). As for the cloud standardization, once again, there is no consensus-based standard framework either for defining the trust itself or for its evaluation.

Based on the discussions in previous sections, the following conclusions can be made:

- there are many published trust characteristics grouped in various cloud trust models, however, they present the points of view of the researchers who developed them. Additionally, most of these trust models pay special attention to security, while without the other aspects of trust, the assessment of trust in the cloud concept would not be complete;
- there are several standards related to cloud computing, however, the level of their completeness and maturity is not stable yet;

- there are several standards presenting software quality models (and characteristics and measures associated to them). These models represent high completeness and maturity, but their applicability to cloud computing technology has not been fully verified yet.

From the perspective of the cloud service provider and cloud service customer, it would be profitable to identify the minimal set of common characteristics that would constitute the basic content of any future cloud computing trust model. In order to identify such potential minimal set of characteristics in course of this research, three sub-phases were executed:

- analysis and concatenation of cloud computing trust-related characteristics present in ISO/IEC standards (table 3-3);
- analysis and concatenation of cloud computing trust-related characteristics present in published research and papers (table 3-4);
- identification of cloud computing trust-related characteristics shared by both areas (table 3-5).

Table 3-3. Cloud computing trust-related characteristics extracted from analyzed ISO/IEC standards

Cloud Computing Trust-related Characteristics Extracted from Analyzed ISO/IEC Standards			
Effectiveness	Context Coverage	Capacity	Maturity
Efficiency	Context Completeness	Compatibility	Fault Tolerance
Satisfaction	Flexibility	Co-Existence	Recoverability
Comfort	Functional Suitability	Appropriateness Recognisability	Usefulness
Freedom from Risk	Functional Completeness	Learnability	Non-Repudiation
Economic Risk Mitigation	Functional Correctness	User Error Protection	Authenticity
Health and Safety Risk Mitigation	Functional Appropriateness	User Interface Aesthetics	Maintainability
Environmental Risk Mitigation	Resource Utilization	Accessibility	Modularity
Reusability	Analyzability	Modify ability	Testability
Portability	Adaptability	Insatiability	Replicability
Suitability	Completeness	Correctness	Appropriateness
Consistency	Courtesy	Traceability	IT Service Reliability
Continuity	Tangibility	Visibility	Professionalism
IT Service Interface Appearance	IT Service Recoverability	Responsiveness	Timeliness
Reactiveness	IT Service Adaptability	Customizability	Initiative
IT Service Maintainability	Accuracy	Credibility	Currentness
Compliance	Precision	Understandability	Broad Network Access
Measured Service	Multi-Tenancy	On-Demand Self Service	Rapid Elasticity and Scalability
Resource Pooling	Governance	Maintenance and Versioning	Protection of PII
Regulatory	Resiliency	Reversibility	SL and SLA

Table 3-4. Cloud computing trust-related characteristics adapted from the analyzed literature

Cloud Computing Trust-related Characteristics Extracted from The Analyzed Literature			
Controllability	Service Use Factors	Assurance	Mobile Access
Company Scale	Data Transmission	Data Processing	Identity
Data Storage	Geographic Situation	Operational Stability	Scalability
Data Privacy	Turnaround Time	Interaction Revolution	Network Speed
Data Security	Turnaround Efficiency	Customization	Success Rate of Transactions
Agility	Transaction Amount	Cost	
Ease of Use	Transaction Number	Payment Flexibility	

Table 3-5. Cloud computing trust-related characteristics shared between published research and ISO/IEC standards

Cloud Computing Trust-related Characteristics Shared Between Published Research and ISO/IEC Standards	
Usability	Time Behavior
Performance	Interoperability
Operability	Reliability
Availability	Security
Integrity	Accountability
Auditability	Confidentiality

Looking at the set of characteristics from table 3-5 (the shared set) it can be noticed that several characteristics that are intrinsic to cloud services are not mentioned either in this set or even in software quality and cloud standards. These trust characteristics are recommended be used as complementary to existing characteristics from broadly acceptable software and systems quality standards and cloud computing standards.

The resulting proposed set of key trust characteristics for cloud computing is presented below.

3.3 The proposed key trust characteristics in cloud computing

According to the analyzed research papers and ISO/IEC standards trust comprises many characteristics, but identified set of commonly recognized characteristics (table 3-5) can reflect the broad, close to a consensus view on main aspects of cloud computing. Hence, for assessing trustworthiness of a cloud service provider, these characteristics could play a significant role as the essential trust characteristics. However, in the contemporary world convincing the users, especially the corporate industrial consumers, to broadly employ cloud services requires a more exhaustive list of characteristics [19].

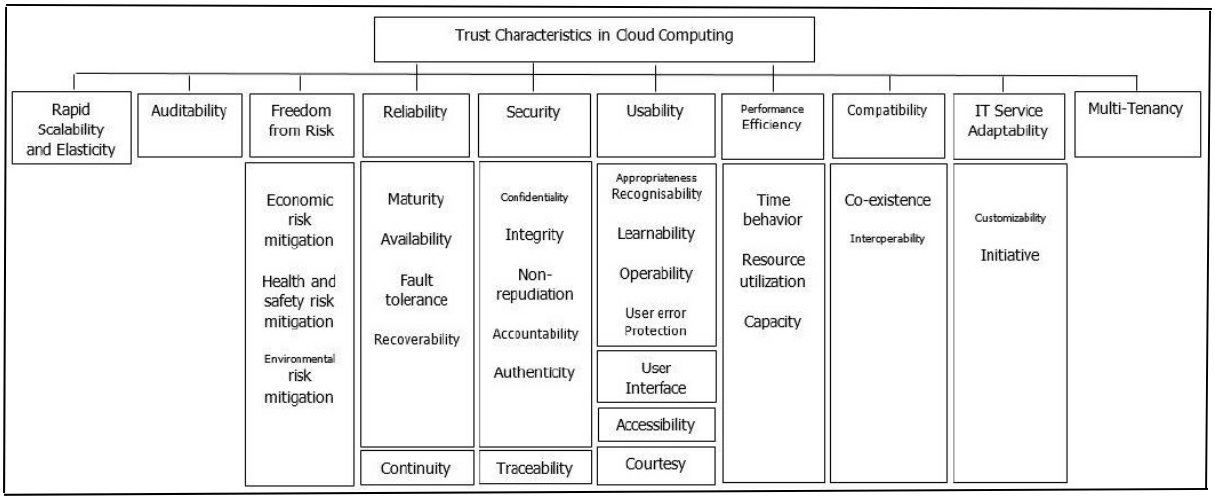

Figure 3-4. The proposed set of trust characteristics in cloud computing

Figure 3-4 presents the proposed characteristics for cloud computing trust. These characteristics are categorized into 10 characteristics, with some of them being further decomposed into sub-characteristics. From table 3-2, it can be induced that scalability, elasticity and multi-tenancy are being considered the intrinsic characteristic of cloud service, and as such were included in the model. It can be noticed that several cross-cutting aspects from table 3-2 were also included in the model. Next, all characteristics/sub-characteristics found as commonly recognized by both academia and standardizing organizations (table 3-5) make the central part of the model. Finally, the quality in use from ISO/IEC 25010 model with its key trust characteristic of freedom from risk (with all sub-characteristics) is also considered a crucial component of the model.

As cloud computing technology is not only relatively new, but also evolving rapidly the gap between the cloud service providers' offers and the mechanisms of controllability available to users is still very large, so the trust rewarded to providers is not based on verifiable measurement but rather on opinions and beliefs.

It is necessary to mention that these results have been already published and remotely presented in a conference in Software Quality Engineering in London, England (SQM 2018) [20].

3.4 Conclusion

As a step towards the development of a broadly acceptable cloud trust model, a literature review on trust characteristics was conducted to identify trust characteristics that were proposed in the domain-related research papers (chapter 1) and those that are published in the broadly acceptable software and systems quality standards and cloud computing standards (the current chapter).

In this chapter which is related to the first stage of matching phase of the research methodology, in order to propose the key trust characteristics for cloud services, software quality and cloud-related standards trust characteristics and these commonly used and

scientifically supported for trust evaluation of cloud service providers were identified. In the first stage of matching both sets of characteristics, the common set of characteristics was identified. Finally, the combination of the identified common set of trust characteristics with these from ISO/IEC 25010 and several cloud computing intrinsic characteristics proposed in published research led to the proposition of the minimal prototype model for cloud services trust.

As the proposed model is minimal and a prototype, the continuation of the research that is supposed to develop a complete and industry and academia consensus-based model, will be explained in the next chapters. Finally, as even the best model that has no measures attached to it is just a theoretical exercise, next chapter will concentrate on adding and identifying required measures.

CHAPTER 4

A PROPOSITION FOR TRUST CHARACTERISTICS MEASURES

4.1 Introduction

A large and growing body of the literature has investigated the role of cloud computing in facilitating data management, sharing hardware and software resources and having a protected interaction between cloud service users and cloud service providers. During the past 20 years, much more information has become available on different requirements and challenges in cloud computing. This shows an increasing awareness for the necessity of evaluating the trustworthiness of cloud service providers. Much of the current literature on trust and trustworthiness pay particular attention to evaluate the level of trust in cloud environments. This evaluation is principally done based on some selected characteristics that mainly have direct relationships with users' requirements. However, there is not any criteria or consensus-based standard to designate these characteristics precisely.

As presented in the first phase of the methodology (analysis phase) and discussed in chapter 1 (literature review), previous studies of trust characteristics have not dealt with the measures of these characteristics comprehensively. Thus, we have recognized that a work on proposing the measures of the trust characteristics for evaluating cloud services according to the system and software quality standards and cloud computing standards is still a necessity. Following the obtained results in which trust characteristics were introduced and presented in chapter 3, in this chapter which is dedicated to the second stage of matching phase of the research methodology, we adapt the extracted trust characteristics from both system and software quality standards and cloud computing standards (details in chapter 3), for evaluating cloud services. Moreover, we derive measures for each trust characteristic to evaluate the trustworthiness of different cloud service providers, and generalize these trust measures for Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Our work thereby demonstrates a way to apply generalized trust measures for cloud services and therefore contributes to a better understanding of cloud services to evaluate their

quality characteristics. As part of our ongoing research, the results of this chapter will be used to develop a comprehensive cloud trust model (ED-BeCT).

4.2 The Main Characteristics Measures to Evaluate Trust in Cloud Computing

In the literature referred to in this chapter, the terms ‘measure’ and ‘metric’ are used interchangeably to clarify the important characteristics which can affect trust in cloud environments. Although they are related, there are some differences. The term ‘measure’ refers to a particular, determined observation of a procedure, whereas, ‘metric’ is defined as the quantifiable elements which are the results of measuring [152]. These quantifiable elements are most commonly known as the ratio, percentage or the number. The specific goal of this study is to present the basic set of trust characteristics measures in conformity with cloud computing standards.

Regarding the proposed key trust characteristics presented in figure 3-4 in chapter 3, in this chapter, we present a holistic way of measuring trust characteristics in cloud environments which allows cloud service users to evaluate subjective (e.g. the user`s previous experience) and objective (e.g. current functionality of the cloud services) trust in cloud computing. In addition, we generalized these trust measures for the three types of cloud service models (Software as a Service, Platform as a Service, Infrastructure as a Service) to create a comprehensive reference for evaluating cloud services proposed by cloud service providers. It should be stressed here that the definitions of the trust characteristics and sub-characteristics are mostly extracted from the both cloud computing standard (ISO/IEC 17788 [24]) and system and software quality standards (ISO/IEC 25010 [1] and ISO/IEC 25011 [148]).

4.2.1 Multi-Tenancy (C4.1)

Resource sharing has a pivotal role in cloud computing. Although, economically it has several significant benefits, it brings various challenges as well. Generally speaking, to provide resource sharing in cloud environments, it may be difficult to distinguish the shareable

resources from the isolated ones in cloud environments. As resource sharing is one of the main principles of multi-tenancy, therefore, cloud service providers should be able to offer multi-tenant cloud services that support this principle. Multi-tenancy is one of the key characteristics of cloud computing and it is defined in ISO/IEC 17788 [24] as :

> « a feature where physical or virtual resources are allocated in such a way that multiple tenants and their computations and data are isolated from and inaccessible to one another. Typically, and within the context of multi-tenancy, the group of cloud service users that form a tenant will all belong to the same cloud service customer organization. There might be cases where the group of cloud service users involves users from multiple different cloud service customers, particularly in the case of public cloud and community cloud deployments. However, a given cloud service customer organization might have many different tenancies with a single cloud service provider representing different groups within the organization. »

According to Fehling et al. in [153], the three principles of multi-tenancy which are shared components, tenant-isolated components and dedicated components [154] have to be taken into account to design cloud based applications. As discussed in [153], the main difference is the isolation degree between tenants which is enabled by these patterns. Therefore, isolation has three aspects [153]:

- The performance required by other tenants should not have any effects on the other performance which is experienced by a tenant (Performance);
- The capacity required by one tenant should not have any effects on the capacity available to other tenants (Capacity);
- The components that belong to a tenant should not be accessible to the other tenants (Confidentiality).

In addition, since the fact of hosting confidential data exposes the system to risks of attacks and security breaches, different researchers have defined multi-tenancy as a security issue in cloud computing [155, 156].

Another important factors that can impact multi-tenancy are scalability and elasticity. Scalability and elasticity refer to quickly increase or decrease resources [24]. In a multi-tenant environment that there are shared resources among tenants, a tenant may require one or more resources which are being used by another tenant in different locations. These situations can impact multi-tenancy requirements [157]. Also, the greater scalability and elasticity provided, the better resource sharing requirements can be addressed [156].

Therefore, there is a relatively good correlation between multi-tenancy and performance efficiency, security and scalability and elasticity. Accordingly, as scalability and elasticity in ISO/IEC 17788 [24] are defined as adjusting virtual and physical resources, it can be found that multi-tenancy can be measured by evaluating performance efficiency, security, scalability and elasticity of which their measures are discussed in the following subsections.

4.2.2 Performance Efficiency (C4.2)

Many scholars hold the view that performance efficiency is one of the influential characteristics of cloud services. Accordingly, various researchers evaluate cloud services performance with different measures and perspectives. For example, Villalpando et al. in [158], introduced a three-dimensional model for measuring performance in cloud computing. The authors in this article proposed the three sub-characteristics of performance efficiency along with the sub-characteristics of reliability in ISO/IEC 25010 as the measures to evaluate performance in cloud environments. However, the paper, makes no attempt to provide information on the measures of these sub-characteristics. On the other hand, the research in the subject which is done by Ataş et al. in [159] presented the limited number of performance metrics as follows:

- MFLOPS (Millions of Floating Point Instructions per Second) that is the number of floating point operations performed by CPU per second;
- MOPS (Millions of Operations per Second);
- Response Time;
- Average total operation time for a single record;
- Average total operation time of 100 records simultaneously;
- Memory bandwidth.

Regarding the cloud service performance explanation which is provided in cloud computing standard (ISO/IEC 17788 [24]);

> « a set of behaviours relating to the operation of a cloud service, and having metrics defined in a SLA. »

It can be noticed that the metrics which are defined in SLA (Service Level Agreement) have contributed to the measuring the performance of cloud services. Referring to SLA standard (ISO/IEC 19086-1 [150]), it can be deduced that the important characteristics to assess performance of the proposed cloud services are [150] :

- cloud service response time (cloud service response time observation, cloud service response time mean, cloud service response time variance);
- capacity (number of simultaneous cloud service connections, limitations of available cloud service resources, cloud service throughput, cloud service bandwidth);
- elasticity (speed, precision).

To identify measures for evaluating performance in cloud environments, we conducted a 3-stages procedure. Firstly, the important characteristics for cloud service performance mentioned in SLA standard (ISO/IEC 19086-1 [150]), are mapped to the performance efficiency characteristics defined in ISO/IEC 25010 [1]. This mapping is performed by considering the definition of performance efficiency as a quality characteristic with three sub-characteristics of time behaviour (SC1), resource utilization (SC2) and capacity (SC3)

(ISO/IEC 25010 [1]). Secondly, the measures for performance characteristics and its sub-characteristics defined in ISO/IEC 25023 [160] are extracted. As discussed in this standard (ISO/IEC 25023), the measures for three sub-characteristics of performance efficiency are:

- time behaviour (SC1): mean response time (adapted from ISO/IEC 25023) (m1), response time adequacy (adapted from ISO/IEC 25023) (m2), mean turnaround time (adapted from ISO/IEC 25023) (m3), turnaround time adequacy (adapted from ISO/IEC 25023) (m4), mean throughput (adapted from ISO/IEC 25023) (m5);
- resource utilization (SC2): mean processor utilization (adapted from ISO/IEC 25023) (m6), mean memory utilization (adapted from ISO/IEC 25023) (m7), mean I/O device utilization (adapted from ISO/IEC 25023) (m8), bandwidth utilization (adapted from ISO/IEC 25023) (m9);
- capacity (SC 3): transaction processing capacity (adapted from ISO/IEC 25023) (m10), user access capacity (adapted from ISO/IEC 25023) (m11), user access increase adequacy (adapted from ISO/IEC 25023) (m12).

Thirdly, elasticity (SC4) which was mentioned in SLA standard [150] as one of the important characteristics for cloud service performance with its two measures of speed (m13) and provision (m14) [150], are considered as the other influential characteristics for performance efficiency.

Speed refers to how fast a cloud service can react to (ISO/IEC 19086-1 [150]) :

- The cloud user requests for reallocation of resources;
- Changing work load in case of manual elasticity and automatic elasticity respectively;
- Precision is defined in ISO/IEC 19086-1 [150] as follows:

> « The precision quantity describes how precise the resource allocation meets the actual resource requirements at a given point in time. In the manual case, precision depends on the granularity of the resource allocation, i.e., the minimum amount of resources

> that can be reallocated. Hence, in the manual case, precision is a technical characteristic of the cloud service that does not require measurements (i.e., no metric is associated with it). In the automatic case, precision refers to the difference between the amount of resources that are allocated and the amount of resources that are actually needed (the optimum state) to cope with a given workload. The actual resource allocation may be over-provisioned (i.e., more resources are allocated that are actually needed), or under-provisioned (i.e. the amount of resources that are actually allocated is not sufficient to cope with the actual work load). As opposed to the manual case, in the automatic case the difference between the allocated and the actually needed amount of resources can be determined by a measurement process and hence imply a metric. »

4.2.3 Security (C4.3)

Despite of the rapid development of cloud computing in many industrial fields in the past decade, cloud environments are still prone to various security risks. Hence, security cannot be ruled out as the first concern in the adoption of cloud computing. Although extensive research has been carried out on evaluating security of the proposed cloud services, more efforts are needed for standardizing the security measures to have a precise evaluation of the proposed cloud services.

Numerous studies have attempted to explain security aspects of cloud environments but there is no attempt to find the main measures of security characteristics according to the cloud computing standards as well as system and software quality standards. For example, Halabi et al. in [161] considered confidentiality, integrity, availability and accountability as security aspects or characteristics in a cloud computing environment. Whereas, Abdel-Basset et al. in [162] evaluated security according to its correlation to other characteristics.

Security in cloud computing standard (ISO/IEC 17788 [24]) is one of the key cross-cutting aspects of cloud computing. As explained in this standard, cross-cutting aspects are behaviours that can have effects on activities, components, and multiple roles, and it is not possible to specifically dedicate them to individual roles or components, hence they are

apportioned issues among these components and roles. Security in ISO/IEC 17788 [24], varies from physical security to application security which encompasses needs such as authentication, authorization, availability, confidentiality, identity management, integrity, non-repudiation, audit, security monitoring, incident response, and security policy management. Regarding the sub-characteristics of security depicted in figure 3-4 (chapter 3), by matching security measures extracted from ISO/IEC 25023 [160], we summarized the measures of security in cloud computing as following:

- confidentiality (SC1): access controllability (adapted from ISO/IEC 25023) (m1), data encryption correctness (adapted from ISO/IEC 25023) (m2), strength of cryptographic algorithms (adapted from ISO/IEC 25023) (m3);
- integrity (SC2): data integrity (adapted from ISO/IEC 25023) (m4), internal data corruption prevention (adapted from ISO/IEC 25023) (m5), buffer overflow prevention (adapted from ISO/IEC 25023) (m6);
- non-repudiation (SC3): digital signature usage (adapted from ISO/IEC 25023) (m7);
- accountability (SC4): user audit trail completeness (adapted from ISO/IEC 25023) (m8), system log retention (adapted from ISO/IEC 25023) (m9);
- authenticity (SC5): authentication mechanism sufficiency (adapted from ISO/IEC 25023) (m10), authentication rules conformity (adapted from ISO/IEC 25023) (m11);
- traceability (SC6): traceable outcomes (ISO/IEC 25011 [148]) (m12).

4.2.4 Rapid Scalability and Elasticity (C4.4)

Scalability and elasticity are defined separately in the literature with various measures and evaluation methods and have been discussed by a great number of authors in the literature. Some significant examples of scalability measures are scalability range, resource scalability rate and cost scalability, and for elasticity measures are usage evolution elasticity, users elasticity speed and mean time to quality repair [163]. Kuhlenkamp et al. in [164] explained that scalability benchmarking measures performance changes before and after a scaling action and elasticity benchmarking measures performance side effects of scaling actions. Due to

numerous technologies to provide elasticity in cloud computing, Coutinho et al. in [165] proposed some metrics based on Physics' concepts to measure elasticity (strain and stress, and microeconomics). Hence, there has been a great deal of confusion in the literature regarding measuring scalability and elasticity in cloud environments. Consequently, referring to cloud computing standard can be a useful guide.

Rapid scalability and elasticity are the key characteristics of cloud computing and in ISO/IEC 17788 are explained as [24] :

> « a feature where physical or virtual resources can be rapidly and elastically adjusted, in some cases automatically, to quickly increase or decrease resources. For the cloud service customer, the physical or virtual resources available for provisioning often appear to be unlimited and can be purchased in any quantity at any time automatically, subject to constraints of service agreements. Therefore, the focus of this key characteristic is that cloud computing means that the customers no longer need to worry about limited resources and might not need to worry about capacity planning. »

This explanation clarifies three significant concepts. First, scalability and elasticity have a resource-related concept in cloud environments and resource adjustment should be done in a minimum time interval (speed). Second, cloud service customer is legitimate to utilize cloud resources according to the service level agreement (resource utilization) without any disorder or irregularity. Third, regarding the definition of capacity in ISO/IEC 25010 [1], there should not be any limitation in resource capacity according to the agreement. It means that there should not be any discrepancy between assigned capacity and the required capacity (precision).

Although, scalability is associated with increasing performance when adding further IT resources and does not pay attention to the deletion of these resources, elasticity focuses on the adding and removing of IT resources to adjust systems' performance immediately in case of changing workload [153]. Hence, it might be possible to consider scalability as a subset of elasticity in cloud environments. Regarding the aforementioned explanations, it can thus be

suggested that performance efficiency with all its measures which was discussed in subsection 4.2.2 in this chapter is a reference sub-characteristic of rapid scalability and elasticity.

4.2.5 Reliability (C4.5)

In the literature, reliability in cloud computing is mostly referred as the ability of a system to provide the required services without failure or interruption [166]. It is also known as the probability of all the applications or data resources to involve in the executing the required services successfully [167]. Regarding these definitions, it can be deduced that fault tolerance is the only measure for reliability in cloud computing. Such expressions are unsatisfactory because they fail to acknowledge the significance of the other measures as mentioned in ISO/IEC 25010 [1] and presented in the figure 3-4 which is related to the proposed trust characteristics in chapter 3.

A more accurate definition of reliability can be found in ISO/IEC 25010 [1] that is :

> « degree to which a system, product or component performs specified functions under specified conditions for a specified period of time. »

In addition as explained in this standard and IEC/TC56 on dependability, dependability can include availability and the other sub-characteristics of reliability presented in ISO/IEC 25010. Therefore, to evaluate the cloud service reliability, the measures adapted from ISO/IEC 25010 [1] and ISO/IEC 25023 for its sub-characteristics are proposed as following [160] :

- maturity (SC1): fault correction (adapted from ISO/IEC 25023) (m1), mean time between failure (adapted from ISO/IEC 25023) (m2), failure rate (adapted from ISO/IEC 25023) (m3), test coverage (adapted from ISO/IEC 25023) (m4);
- availability (SC2): system availability (adapted from ISO/IEC 25023) (m5), mean down time (adapted from ISO/IEC 25023) (m6);

- fault tolerance (SC3): failure avoidance (adapted from ISO/IEC 25023) (m7), redundancy of components (adapted from ISO/IEC 25023) (m8), mean fault notification time (adapted from ISO/IEC 25023) (m9);
- recoverability (SC4): mean recovery time (adapted from ISO/IEC 25023) (m10), backup data completeness (adapted from ISO/IEC 25023) (m11);
- continuity (ISO/IEC 25011 [148]) (SC5): supported cloud services (m12) (The percentage of the supported cloud services. e.g. to mitigate the risks resulting from interruption to an acceptable level).

With regard to the explained measures of reliability and the definition of resiliency in ISO/IEC 17788 [that is ability of a system to maintain an acceptable service level in case of faults which can impact on the normal operation] as one of the cross cutting aspects of cloud computing, it can be said that reliability and resiliency have close correlation. As there is no explanation of resiliency in the quality standards, we consider reliability as a characteristic with the close concept to resiliency. Therefore by assessing reliability as one of the essential trust characteristics, resiliency can be addressed to some extent as well.

4.2.6 Freedom from Risk (C4.6)

As mentioned in the literature, selection of a trustworthy cloud service provider is a decision-making problem. Therefore, making a decision meticulously needs risk analysis, control and mitigation [168]. But, to the best of our knowledge, ‘freedom from risk’ as a trust characteristic to evaluate in cloud environments is significantly overlooked in the literature.

From monetary perspective, cloud computing causes cost saving and reduces expenditures. This significant benefit of cloud technology is a pivotal reason for the organizations to switch to cloud environments [169]. But as return on investment (ROI) is a long-term goal [170], and during this period, many individuals and stakeholders may be attracted and migrate their data to cloud environments, neglecting this characteristic (freedom from risk) can lead to

irreparable damages such as very huge financial losses, disclosing information and confidential data, wasting time and human resources.

In broad terms, risk is the probability of a perilous case occurrence that can have an effect on the achieving goals [171]. Consequence or impact and likelihood of the event are the main factors for risk measuring [172]. As explained in ISO/IEC 25022 [173], inadequacy of any product quality characteristic or inadequate levels of effectiveness and efficiency can cause the risks of undesirable consequences. In addition, risks of undesirable consequences can have impacts on the user of a system, organizations which are using a system, organizations that are developing a system and a wider community [173].

Accordingly, freedom from risk in ISO/IEC 25010 [1], is :

> « degree to which a product or system mitigates the potential risk to economic status, human life, health, or the environment. »

Regarding its sub-characteristics depicted in figure 3-4 (chapter 3), measures for evaluating this characteristic (freedom from risk) are explained as following (ISO/IEC 25022 [173]) :

- economic risk mitigation (SC1): return on investment (adapted from ISO/IEC 25022) (ROI) (m1), time to achieve return on investment (adapted from ISO/IEC 25022) (m2), business performance (adapted from ISO/IEC 25022) (m3), benefits of IT investment (adapted from ISO/IEC 25022) (m4), service to customers (adapted from ISO/IEC 25022) (m5), cloud customers loyal to a specific cloud service provider (m6) (the percentage of the loyal cloud customers compared to all the customers of a specific cloud service provider), revenue from each customer (adapted from ISO/IEC 25022) (m7), errors with economic consequences (adapted from ISO/IEC 25022) (m8);
- health and safety risk mitigation (SC2): user health reporting frequency (adapted from ISO/IEC 25022) (m9), user health and safety impact (adapted from ISO/IEC 25022) (m10), safety of people affected by use of the system (adapted from ISO/IEC 25022) (m11);

- environmental risk mitigation (SC3): environmental impact (adapted from ISO/IEC 25022) (m12).

4.2.7 Usability (C4.7)

Regarding the competitive atmosphere among cloud service providers for attracting customers, usability is a key aspect for addressing users' requirements. Thus, evaluating usability is the hidden characteristic to thrive in this competitive environment [174]. Moreover, since there are different cloud deployment models without stability to the users' experience, it is being remarkably recognized by cloud service users that enhancing cloud usability standards is necessary to certify stability among various cloud services proposed by diverse cloud service providers [175].

Usability is an attribute of a product that measures to what extent it can be usable by different users with efficiency, effectiveness and satisfaction to achieve their determined goals [176]. Also usability refers to the ability of a product to be understood, learned, operated and is habitually verified from its interfaces [177, 178]. More specifically, usability in ISO/IEC 25010 [1], is defined as :

> « degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use »

To evaluate usability, the measures for its sub-characteristics (figure 3-4 from chapter 3) which are extracted from ISO/IEC 25023 [160] explained as following, should be taken into account.

- appropriateness recognisability (SC1): description completeness (adapted from ISO/IEC 25023) (m1), demonstration coverage (adapted from ISO/IEC 25023) (m2), entry point self-descriptiveness (adapted from ISO/IEC 25023) (m3);

- learnability (SC2): user guidance completeness (adapted from ISO/IEC 25023) (m4), entry fields defaults (adapted from ISO/IEC 25023) (m5), error message understandability (adapted from ISO/IEC 25023) (m6), self-explanatory user interface (adapted from ISO/IEC 25023) (m7);
- operability (SC3): operational consistency (adapted from ISO/IEC 25023) (m8), message clarity (adapted from ISO/IEC 25023) (m9), functional customizability (adapted from ISO/IEC 25023) (m10), user interface customizability (adapted from ISO/IEC 25023) (m11), monitoring capability (adapted from ISO/IEC 25023) (m12), undo capability (adapted from ISO/IEC 25023) (m13), understandable categorization of information (adapted from ISO/IEC 25023) (m14), appearance consistency (adapted from ISO/IEC 25023) (m15), input device;
- support (adapted from ISO/IEC 25023) (m16);
- user error protection (SC4): avoidance of user operation errors (adapted from ISO/IEC 25023) (m17), user entry error correction (adapted from ISO/IEC 25023) (m18), user error recoverability (adapted from ISO/IEC 25023) (m19);
- user interface aesthetics (SC5): appearance aesthetics of user interfaces (adapted from ISO/IEC 25023) (m20);
- accessibility (SC6): accessibility for users with disabilities (adapted from ISO/IEC 25023) (m21), supported languages adequacy (adapted from ISO/IEC 25023) (m22);
- courtesy [148](SC7): Polite message adequacy (m23).

4.2.8 Compatibility (C4.8)

Cloud computing enables ubiquitous network access to the shared resources [2], hence, there is a possibility that by producing new opportunities, makes some challenges as well. One of these challenges is compatibility with customers' components and services; i.e. the proposed cloud services must be compatible with the customers' requirements, frameworks, policy and environment [169, 179] as each customer has different requirements.

On the other hand, regarding the definition of compatibility in ISO/IEC 25010 in which the product or system should be able to exchange data through the same environment, it can be deduced that aligning the offered services with the Internet platform is the other aspect of compatibility in cloud computing. As it is believed that by the alignment of the cloud computing platforms through Internet platform more precisely, the capacity to take the advantages of cloud computing will be enhanced and also the level of skepticism among the individuals and organizations for using cloud services, will be reduced significantly [180]. Because, using the cloud resources and services through Internet remotely can facilitate the users' business procedures remarkably.

From users' perspective, in case of using a single cloud service which is not able to completely address users' needs, inevitably, combining cloud services from different cloud providers is an alternative [181]. But checking the compatibility of the other cloud services proposed by various providers can be a challenging task particularly, for non-expert users [181]. Therefore, to evaluate the compatibility of a cloud service, as defined in ISO/IEC 25010 [1] as :

> « degree to which a product, system or component can exchange information with other products, systems or components, and/or perform its required functions, while sharing the same hardware or software environment. »

and regarding its sub-characteristics depicted in figure 3-4 (chapter 3), its measures are extracted from ISO/IES 25023 [160] can be presented as following:

- coexistence (SC1): coexistence with other cloud services (adapted from ISO/IEC 25023) (m1) (to what extent the determined cloud service can share the environment with other cloud services without adverse impact on their quality characteristics or functionality);
- interoperability (SC2): data format exchangeability (adapted from ISO/IEC 25023) (m2) (the proportion of the specified data formats is exchangeable with other cloud services), data exchange protocol sufficiency (adapted from ISO/IEC 25023) (m3), external interface adequacy (adapted from ISO/IEC 25023) (m4).

4.2.9 IT Service Adaptability (C4.9)

In ISO/IEC 25011 [148], IT service adaptability is defined as :

> «degree to which an IT service can configure itself or be modified to meet new needs. »

In terms of time, having new needs refers to the fact that there are two types of requirements;

- actual requirements: the determined requirements of the users before using cloud services;
- future requirements: the requirements that users will come across deliberately or accidentally after the adoption of a cloud service proposed by a cloud service provider.

Therefore, as discussed in ISO/IEC 25011 [148] and depicted in figure 3-4 (in chapter 3), customizability and initiative are the main sub-characteristics of IT service adaptability. To cover future requirements, the proposed cloud service must be adaptable and to ensure the IT service adaptability, it should be measured. Regarding the three well-known service models in cloud computing (i.e. Software as a Service, Platform as a Service, and Infrastructure as a Service), hardware and software adaptability of the proposed cloud services should be considered. These are the essential aspects of trust in cloud environments that can address different user's requirements. The important measures of this characteristic are as following [160] :

- customizability (SC1): functional customizability (adapted from ISO/IEC 25023) (m1), user interface customizability (adapted from ISO/IEC 25023) (m2);
- initiative (SC2): Initiative adequacy (m3).

4.2.10 Auditability (4.10)

Auditing is generally seen as a characteristic strongly related to assure users that the proposed cloud services have the acceptable quality (or not). In addition, the results of auditing can be

an important determinant in the cloud service provider selection. In the literature, there are numerous ways for auditing cloud services and cloud service providers. However, a closer look at the literature on the contributory characteristics to conduct a rigorous auditing, reveals a number of gaps and shortcomings.

Referring to ISO/IEC 17788 [24], auditability is :

> «the capability of collecting and making available necessary evidential information related to the operation and use of a cloud service, for the purpose of conducting an audit. »

Therefore, based on the aforementioned explanations, it can be simply justified that evaluating the auditability of the proposed cloud services stems from evaluating the necessary characteristics of cloud services which are explained in previous subsections.

4.3 The dependencies and correlations of the trust characteristics

Basically, the trust characteristics presented in the literature, are related to the principal aspects of cloud computing. However, the measures we found in the literature for assessing the proposed trust characteristics do not conform to the system and software quality standards, nor are they easily applied to the context of cloud computing. Since there is no rule to select cloud trust characteristics measures, we chose instead to refer to existing standards and identify these measures. We chose these standards as they are agreed-upon reference, available, and have scientific and academic support.

On the other hand, to date, the existing literature on trust characteristics and their measures are mostly based on the authors' perspectives by their analysis of the literature. But this approach may not be comprehensive enough in order to evaluate trust in cloud environments, since there may be some inconsistencies and discrepancies while proposing trust characteristics and their measures [182, 183].

The present chapter aimed to determine the measures of trust characteristics based on analyzing standard characteristics of cloud computing and IT services, extracted from system and software quality standards and cloud computing standards. Figure 4-1, illustrates the trust tree of cloud computing along with the standard measures which can clearly clarify the dependencies and existing correlations among the proposed characteristics. In addition, this trust tree is complementary to the results of which are depicted in figure 3-4 in chapter 3.

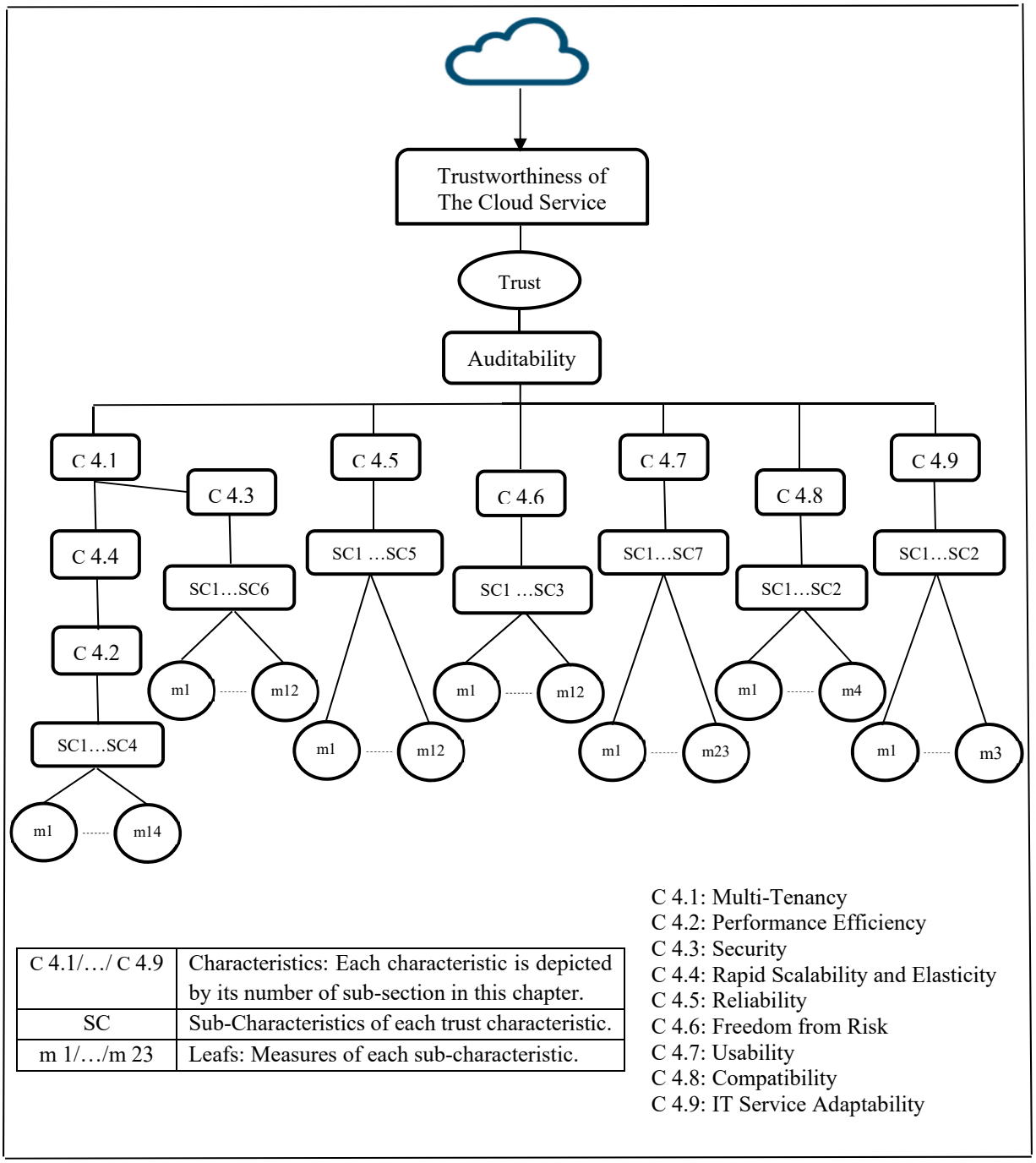

Figure 4-1. The trust tree of cloud computing

As depicted in figure 4-1, auditability is placed as the sub-layer of trust which is the measure of trustworthiness in cloud computing. We decided to place it here since the evaluation of cloud services by cloud providers may be complicated by dishonesty on the part of the service providers [12]. This can be the main problem in the cloud computing adoption. Therefore, outsourcing the auditing cloud services to a trusted third party can produce more trustworthy results. Taking into account this fact, it can thus be concluded that by outsourcing auditing procedures, users are searching for a way to boost their confidence about cloud service provider selection and to ensure that the selected cloud services can cover the determined requirements. In the light of these principles and regarding the definition of trust in the system/software quality standard (ISO/IEC 25010 [1]), namely;

> « the degree to which a user or other stakeholder has confidence that a product or system will behave as intended. »

the correlation between trust and auditability can clearly be justified. Moreover, auditability of the system can be assumed as a gateway in its evaluation. If this gateway does not exist in the system, there is a possibility that the system may be unavailable for any form of real, fact-based evaluation.

The presented trust tree can be considered as complete as possible trust characteristics reference in the time of conducting this research, along with the associated measures in cloud environments. What remains is for the proposed trust characteristics measures to be applied to real scenarios in order to assess their applicability in evaluating the trustworthiness of the cloud service provider. The presented measures can be used to evaluate both subjective (e.g. the user`s previous experience) and objective (e.g. current functionality of the cloud services) trust in cloud environments.

4.4 Conclusion

The literature has highlighted several methods to define measures for evaluating trust characteristics in cloud computing. However, all the previously mentioned methods suffer

from some serious shortcomings that we aimed to address in this chapter. First, we cannot neglect the contribution of the existing related ISO/IEC standards when establishing measures for cloud trust characteristics. Second, there are various methods proposed by different researchers that they may not be completely applicable to evaluate trust characteristics in cloud environments, since they lack scientifically supported measures. Third, because the notion of trust also has uncertainty associated with it, by combining both subjective and objective evaluation of trust in cloud environments we can achieve a more credible outcome.

We investigated cloud computing standards as well as system/software quality standards in particular ISO/IEC 25000 series to extract measures for evaluating the proposed trust characteristics in cloud computing (which is also the topic of the current research activity of the SC7 WG06 study group to verify the applicability of the quality standards to cloud computing) as all the definitions in these standards are expanded from software to the systems which cloud technology can be considered as a combination. We reviewed the literature on evaluating these characteristics in cloud environments to conform the proposed measures to various perspectives. The result of this conformity is depicted as a trust tree to be considered as a reference for evaluating trust in cloud environments.

It should be noted that this chapter of the current research project is already published in the journal of computer and information science in [14].

Moving forward, a natural progression of these results is to propose a cloud trust model to evaluate the proposed trust characteristics according to their identified measures. But before that, in the next chapter, we will begin the first stage of the tracing phase in the research methodology which is related to IaaS, PaaS and SaaS features to be able to verify the applicability of the identified measures.

CHAPTER 5

THE FIRST STAGE OF TRACING PHASE : AN INVESTIGATION INTO IAAS, PAAS, SAAS

Cloud computing is an efficient way of computing in which the computational resources either software-based or hardware-based or even both types concurrently are managed automatically and delivered to the users through the Internet [184, 185]. Thus, regarding the users' requirements, cloud computing is divided into three service models [2, 186] : Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service(SaaS). Although, there are the other cloud service categories as explained in ISO/IEC 17788 such as Compute as a Service (CompaaS) and Data Storage as a Service (DSaaS), it can be said that SaaS, PaaS and IaaS are the core service models that are mostly considered in the literature. Cloud service models can be offered through different cloud deployment models which are public cloud, private cloud, community and hybrid cloud [2]. Generally speaking, cloud service models and cloud deployment models are the two points of view on cloud provider [153] :

> « Cloud service models describe the style how IT resources are offered. Cloud deployment models describe the cloud environments hosting these IT resources, especially, regarding the group of customers they are made available to. Therefore, a combination of cloud service model and cloud deployment type characterizes the environment of a cloud provider. »

Accordingly, cloud service models and cloud deployment models are substantially connected [153]. Figure 5-1 presents this concept more clearly. Thereby, it seems possible that cloud trust models are applicable in various cloud deployment models.

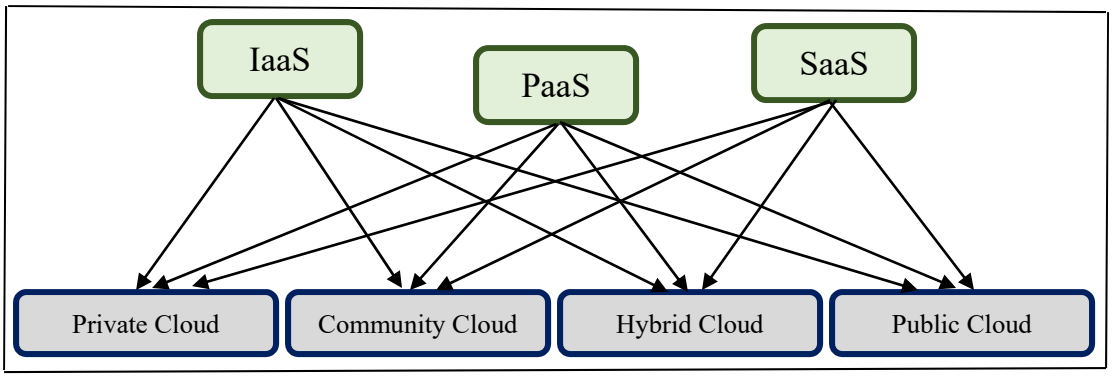

Figure 5-1. Connection between cloud service models and cloud deployment models
Taken from [153]

In the chapter that follows, we provide an overview of cloud service models, their characteristics and their applicability in cloud computing. The purpose of this investigation is to explore the existing similarities and the differences between IaaS, PaaS and SaaS, to be able to verify the applicability of the trust characteristics in cloud environments correctly.

5.1 Cloud service models

To compare and categorize cloud service models, various layers of the application stack are considered [153]. In broad terms, application stack can be defined as a set of application programs include software applications that assist in the performance of a task. There is a close link among these applications and data export and data import can be done among them in a short time. An application stack by providing application programs, facilitate the workflows and provide the work environment. An application stack comprises six layers [153] :

- Physical hardware – physical infrastructure that contains servers, storage, networks connecting servers, racks containing servers, power lines, etc.;
- Virtual hardware – a hypervisor and virtual networking can abstract and map the physical hardware components to virtual counterparts to enable sharing physical hardware between virtual counterparts. As an example, mapping virtual servers to some physical servers can ensure the users of the virtual servers perceive they were the only users who can have access to it, but in fact, physical hardware is shared. So, virtualization enables a single physical

resource serves as multiple virtual resources or vice versa, provide multiple physical resources function as a single virtual resource [187]. In addition, in a datacenter, in case of having some changes in configurations, virtualization can significantly reduce the adjustments of tangible physical resources, i.e. servers, switches, cables, etc.

Hence, virtualization is a component that through cloud computing can separate operating system from hardware on which it is performing [187]. Thus, with virtualization we can run two environments on the same machine completely isolated [188].

Virtualization layer which is known as hypervisor provides hardware resources (such as CPU, Memory, Network, etc.) and resides between hardware and the virtual machine on which an operating system is running [187]. There are two types of hypervisor [187] : type 1 that is known as bare-metal hypervisor which is installed in hardware and the operating system can be installed on the top of that hypervisor – type 2 that is known as hosted hypervisor, in this type an operating system is installed on the hardware, then the hypervisor is installed on that operating system. The instance of operating system can be installed over that hypervisor.

- Operating system – on physical or virtual hardware, software will directly be installed. Operating systems by providing functions to applications which are installed on them can abstract hardware. Multitasking operating systems and virtualization have some similar capabilities such as providing various virtual servers centralized in a single physical machine [187];
- Middleware – in [189] middleware is defined as a software which resides between applications, services and their underlying distributed architecture and platforms. In addition,

> « Middleware provides several types of capabilities to developers, including providing higher-level programming abstractions to support the development of applications and services; supporting end-to-end quality attributes, such as scalability, persistence, and security; and masking problems such

> as system failure and heterogeneity of languages, operating systems, and networks [189]. »

Software on this layer will be installed over an operating system and an environment to install and execute custom applications is provided by itself. Moreover, middleware can provide communication services and handle data storage such as MySQL and Oracle [153];

- Application software – this layer associate with the custom applications with provided functionality to users or other applications such as customer relationship management systems (CRM) and enterprise resource planning (ERP);
- Business process – this layer is related to a set of applications that support the processes of an enterprise.

Regarding these six layers of application stacks, cloud service models can reside in any of these layers [153]. Figure 5-2 represents the schematic of a mapped application stack to the cloud service models.

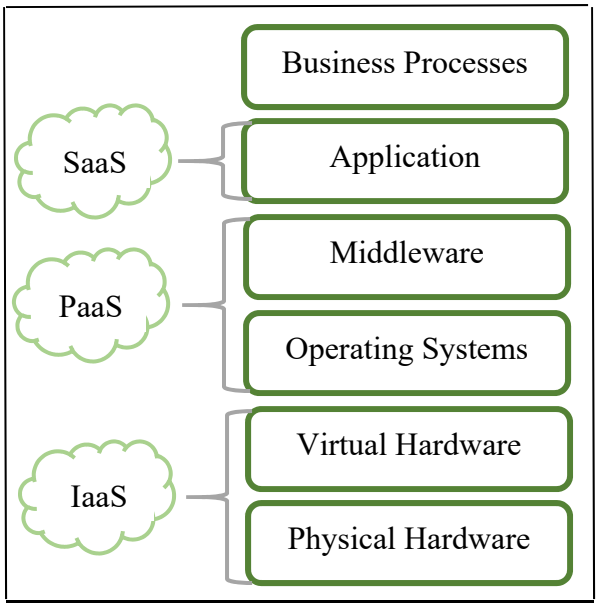

Figure 5-2. Application stack mapped to the cloud service models
Taken from [153]

5.1.1 Infrastructure as a Service (IaaS)

Infrastructure as a service providers offer a combination of hardware such as servers, storage and network and related software such as operating systems virtualization technology and file system to assist users to fulfill their tasks [190]. Thus, to accomplish the users' objectives, the computing resources of the IaaS must be managed efficiently and there is a need to schedule the tasks reasonably to reduce the makespan (the maximum time to a task completion) of the tasks [191]. However, the customer is responsible to monitor and ensure the proper operation of all the customized software and licensed software [190]. The IaaS provides a flexible, accessible and scalable infrastructure to deploy a large-scale scientific workflow [192]. As explained in [192]; a workflow refers to a set of computational tasks which there are some dependencies between them and scientific workflows that are managed by various organizations or individuals in diverse domains, indicate that they have various needs for the software required by the tasks to run.

On the other hand, there are some issues related to the IaaS in cloud systems which are briefly introduced as follows [193] :

- Virtualization and multi-tenancy: despite the fact that hardware virtualization is beneficial, there are some problems in high-level scalability which is required for providing cost-effective cloud computing for masses. It is believed that multi-tenant virtualization by focusing on software virtualization can be a helpful solution to remedy these issues. The other major issue is in network virtualization that refers to the resource usage in network layer efficiently and avoid congestion in this layer;
- Resource management: as computational resources are limited, an efficient resource allocation should effectively handle various workload fluctuations and guarantee quality of services. Thus, to have an efficient resource management the issues such as resource mapping, resource provisioning, resource allocation as well as resource adaptation should be taken into consideration. So, to provide elastic pricing and billing it is necessary to meter any resource and service consumption;

- Network infrastructure management: to manage a wide range of network components such as hubs and bridges, automated methods are required for common system management tasks. These automated methods have to handle the increased monitoring data size which is higher than current systems;
- Security, privacy and compliance are fundamental in the systems that deal with sensitive data and code. Therefore, it is necessary to ensure the adequacy of several security issues such as authentication, data confidentiality, integrity and nonrepudiation;
- Data management which is a vital aspect especially for storage in cloud environments where data can be flexibly distributed through multiple resources. Furthermore, it is required to maintain data consistency in a broad replicated data sources distribution. In addition, to consider taking latency and workload, the location of data would be pivotal in case of replication across data centers;
- APIs and/or programming developments are necessary to benefit from cloud computing. Thus, a cloud environment should be able to provide the features in a way that allows the user not to be worried about scalability and autonomic capabilities and deposit these types of management to the system;
- Tools are essential to support development, adaptation and usage of cloud services. Cloud computing can reduce the challenges of getting a final product by providing tools and processes to exploit a complete server and storage without requiring to have technical specialists. The issue is in developing tools that can produce accurate results.

From the customers' perspective on IaaS services [190] :

- These services provide the capabilities for the users to access applications from anywhere;
- It is a modular, flexible, scalable, virtualized and automated system;
- They are always available and resilient;
- They provide the capabilities to put the applications and data on provisioned platform and maintained by the provider.

From providers' perspective on IaaS services [190] :

- They are responsible to provide virtual infrastructure such as servers, storage, network virtualization;
- They are in charge of provisioning of space, power and cooling;
- Deployment of web base applications to provision infrastructure for customers if needed;
- Providing load balancing services;
- Facilitate the process of application propagations on infrastructure instances;
- Making the infrastructure services available for the users;
- Preserving the security of CPUs, data and network;
- Managing account and provisioning.

5.1.2 Platform as a Service (PaaS)

PaaS provides a development platform that supports the software lifecycle completely and enables users to develop cloud services and applications such as SaaS directly on the PaaS cloud [194]. Basically, PaaS provides mechanisms for deploying applications, designing cloud applications, pushing applications to their deployment environments, using services, database migration, mapping custom domains, IDE plugs and a build integration tool [195]. Thus, there is no need for the users to buy and install software and hardware to deploy an application [196].

PaaS supports the applications of all tenants and can address the multi-tenancy issues [197]. Also, it has built-in fault-tolerance mechanisms and offers scalable computing [197]. As described in [198], there are three groups of PaaS that can be placed between IaaS and SaaS:

- IaaS-centric PaaS that enable deployment of application over the IaaS stack and in the meanwhile provide a full control of underlying infrastructure;
- SaaS-centric PaaS that focus on productivity and simplicity tailored to a particular SaaS solution. These platforms are able to abstract the available middleware and without changing the application code, they can be composed through visual tooling help. Some

particular platforms with the purpose of Business Intelligence or Business Process Management are included in SaaS-centric PaaS;

- Generic PaaS that supply a classical application platform with a set of language runtimes, frameworks, services and other components required to program an application. This is the PaaS provider who is responsible to manage the platform. Whereas, the essential aspects such as scaling can be controlled through a management interface by developers.

5.1.3 Software as a Service (SaaS)

Basically, SaaS is one of the cloud service models and the top layer of cloud computing that supports multi-tenant architecture which enables users to share the same infrastructure on which the software and related data are hosted and accessed by the users through Internet and a web browser [185]. SaaS provides a single application which its configuration can be customized by several customers [185].

In this layer, the users are able to benefit from the software with the SaaS provider's license and pay for that on the pay-per-use mode [185]. Users can select pre-defined applications whereas SaaS providers are responsible to manage and control the hardware and software infrastructure regarding the essential aspects such as scalability [199]. As explained in [200] :

> « Cloud consumers do not have control over the cloud infrastructure that often employs multi-tenancy system architecture, namely, different cloud consumers' applications are organized in a single logical environment in the SaaS cloud to achieve economies of scale and optimization in terms of speed, security, availability, disaster recovery and maintenance. »

Regarding the SaaS capabilities, it can be said that SaaS is different from PaaS in hosting applications. Since SaaS only hosts final cloud applications while PaaS provides a development platform in which the completed and in-progress cloud applications can be hosted [194]. Subsequently, PaaS needs to support application hosting environment as well as

having development infrastructure in which programming environment, tools, configuration management, and etc. are included [194].

5.2 Conclusion

Regarding the aforementioned descriptions, it seems that provisioning of the resources in a flexible and abstracted way would be the main similarity between the three cloud service models [199]. Accordingly, there are three types of resources [199] :

- Computing resources,
- Storage resources,
- Network resources, which are used by/in the three cloud service models.

On the other hand, the key difference between IaaS, PaaS and SaaS could be the access level of the cloud service user and the cloud service provider in terms of each cloud service model. Figure 5-3 can clearly describe this difference. It is apparent from this figure that in terms of SaaS, the cloud users have the least control on the cloud resources. By contrast, it can be seen that data and applications are under control of the cloud users in PaaS as well as IaaS. However, cloud users have the broadest level of access to the cloud resouces by using IaaS in comparison with SaaS and PaaS.

Therefore, it seems that evaluating trust characteristics based on their measures which are discussed in the next chapter, would depend on the level of access in cloud environments which is one of the significant criteria in evaluating the trustworthiness of the cloud service provider and measuring cloud services.

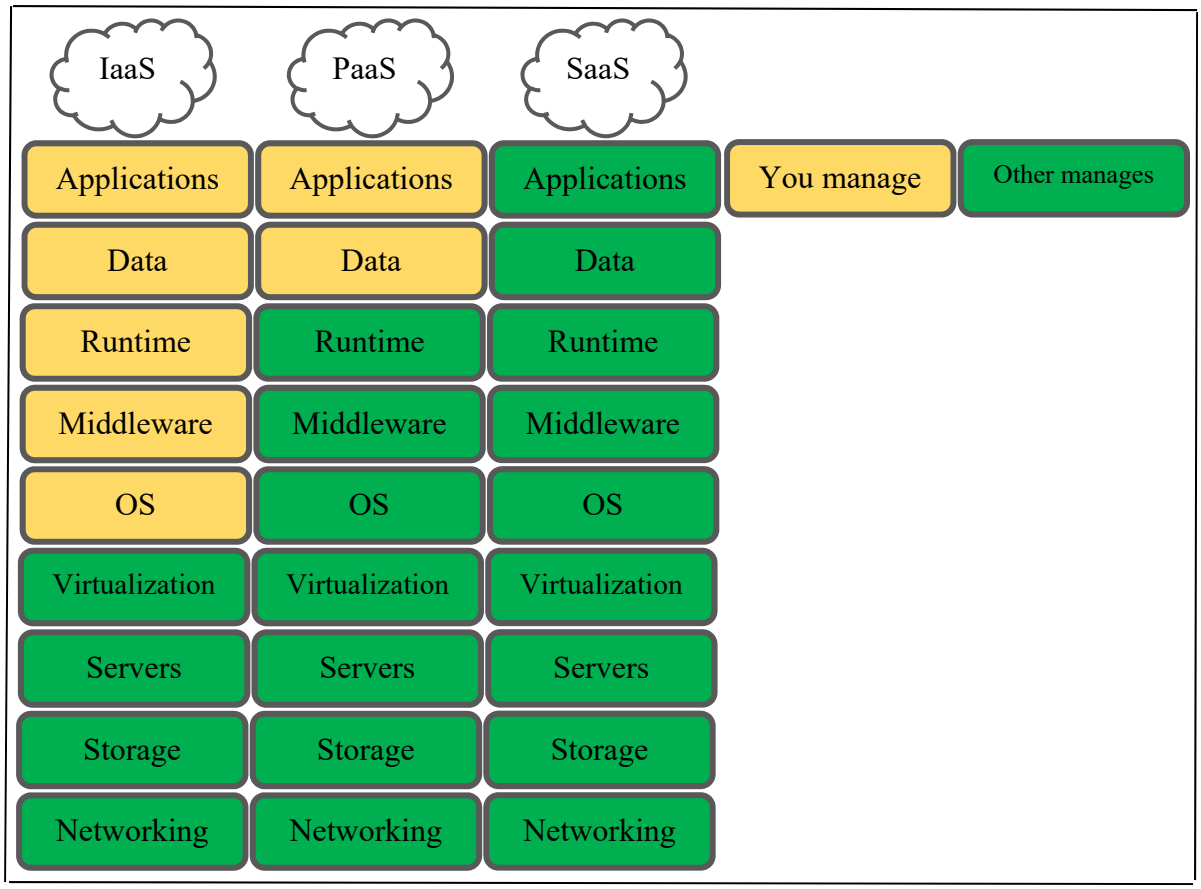

Figure 5-3. The main difference between IaaS, PaaS and SaaS

Taken from [201]

To specify the feasibility of evaluation of the proposed trust characteristics measures according to the aforementioned discussion, the verification stage will be performed in the next chapter. Therefore, in chapter 6 we will present the second stage of tracing phase of the research methodology.

CHAPTER 6

THE SECOND STAGE OF TRACING PHASE : VERIFICATION OF THE APPLICABILITY OF THE IDENTIFIED MEASURES

The quality of the software-based systems can be defined as a collection of software quality characteristics [202] (chapter 3). These characteristics might be categorized into two main categories of static characteristics (such as extensibility, cohesion, complexity) and dynamic characteristics (the ones related to the system behavior during its execution such as throughput, fault tolerance and scalability) [202]. Since cloud computing has a dynamic nature, in trust evaluation process and assessing cloud services evaluating the dynamic characteristics has a pivotal role.

To ensure that a cloud service can address the identified expectations regarding the quality characteristics, it is important to design proper ways to measure them and assure the cloud services satisfy the requirements [202]. As the main goal is to select the high-quality cloud service(s), measurement can be used to evaluate the quality of the cloud services [203]. Further, software metric in [203, 204] is defined as :

> « objective, mathematical measure of software that is sensitive to differences in software characteristics. It provides a quantitative measure of an attribute which the body of software exhibits. »

Measurements can be helpful to estimate quality of cloud services [203]. Whereas, without measurement judgment may be based on subjective evaluation [203]. Indicators or metrics can measure quality indirectly and are able to provide insight into the cloud services [203, 205]. Generally speaking, software measurement is categorized into two categories [203] :

- Direct measurement which includes lines of code (LOC) produced, speed of execution, size of memory, and reported defect over a specified time;

- Indirect measurement which includes performance efficiency, reliablity, etc. These characteristics are unmeasurable quality characteristics. Therefore, to generate a measurable metric, they decomposed into various sub-characteristics.

Subsequently, in order to evaluate the trustworthiness of a cloud provider several cloud trust models paying attention to various characteristics in cloud environments were presented. These characteristics in cloud environments were further considered as the set of “trust characteristics”.

In chapter 4, the measures of trust characteristics by matching the literature with the applicable standards are presented. But according to our findings and regarding the notion of cloud computing, some elements of these measures are still far away from being evaluated in reality. Although, these measures are extracted from system and software quality standards, there are various difficulties or in some cases it is impossible to measure the elements of the proposed measures in cloud environments.

A reasonable approach to tackle this issue could be to verify the applicability of the elements of the measures from the feasibility perspective of their evaluations by their proposed evaluation functions. Therefore, in this chapter we categorized these measures into two groups of “doable” and “undoable”. However, it needs to be emphasized here that a full discussion of the solutions for “undoable” measures, lies beyond the scope of this study. Since each measure requires a separate comprehensive research.

In addition, due to the different levels of access in cloud environments, this chapter attempts to clarify the type of “user” who is supposed to assess the cloud services or to evaluate the trustworthiness of the cloud service provider. Moreover, in some cases due to the complexity, users need to be equipped to be able to measure the introduced elements. Thus, to make the proposed cloud trust model easier to apply and more transparent to use, it is important to specify the type of user. In addition, since the proposed cloud trust model would be useful for the cloud service providers as well to evaluate the provided cloud services, we add cloud

service providers in our categorization as a type of cloud user. In this chapter, regarding the level of access in cloud environments, users are categorized into three groups:

- Individual: the end user of the cloud services;
- Organization: two or more cloud service providers that have collaboration together;
- Cloud service provider (CSP): the owner of the provided cloud services.

Figure 6-1 represents the different access levels to the cloud resources based on the aforementioned categories. As illustrated in this figure, cloud service provider has the broadest level of access to cloud resources in cloud environments since, this is the cloud provider who controls and configures the provided services. After cloud service providers, it is expected that organizations possess intermediate level of access. Consequently, it is almost certain that individuals (end users) have the lowest access level. Therein after, in the next chapter regarding the results of this chapter, a comprehensive cloud trust model will be proposed.

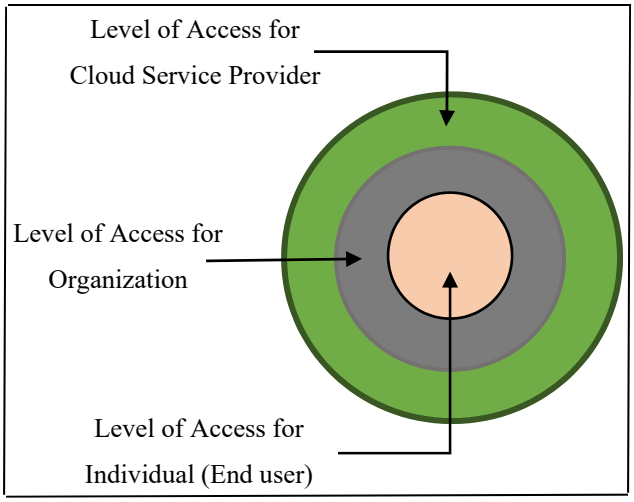

Figure 6-1. Levels of access to the cloud resources for different types of users.

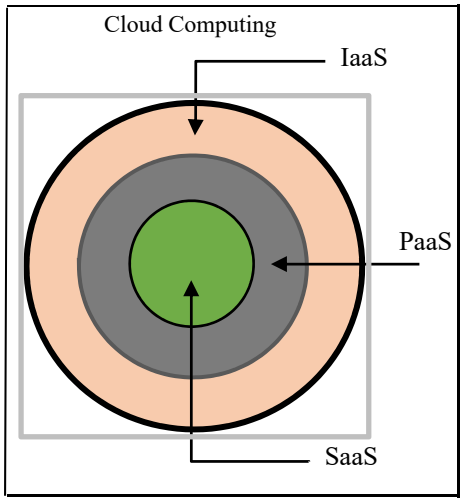

Figure 6-2. The adapted perspective of cloud computing with its cloud service models

In addition, in this research project we adapt a different perspective of cloud computing from the literature to assess the trustworthiness of the cloud service providers. In figure 6-2, the adapted perspective of cloud computing with its three layers of cloud service models is depicted. Regarding figure 5-3 (in the previous chapter), the broadest level of access to the cloud resources for cloud users is provided in IaaS. Then, the level of access in PaaS and SaaS would be more limited respectively. We present this fact by inverting the colors of figure 6-1 in figure 6-2. Consequently, we believe that based on this perspective (figure 6-2), in order to have an accurate evaluation of cloud services, it is essential to consider the three layers of cloud computing outright in the evaluation process. Because, regarding the explanations related to the IaaS, PaaS and SaaS in chapter 6 and based on the fact that SaaS is on the top of PaaS (regarding the application stack figure 5-2 chapter 5), and PaaS is also implemented over IaaS, therefore, they have close interconnections with each other. Subsequently, considering the three layers of cloud computing separately for assessing cloud services (which is a common view in the literature) may not lead to a precise result.

Regarding the adapted perspective (figure 6-2) and the generalized characteristics with their measures for three cloud service models, the second stage of tracing phase of the research methodology is carried out in the current chapter.

6.1 The verification of performance efficiency measures

The aim of the introduced measures for performance efficiency is to assess the performance of the cloud resources (include software products, the software and hardware configuration of the system, and materials e.g. storage media) which are used under the stated conditions (ISO/IEC 25023 [160]). Regarding this fact, performance efficiency is a crucial trust characteristic for SaaS, PaaS and IaaS.

6.1.1 Time Behavior

Table 6-1 presents the measures of this sub-characteristic in detail. Since they are time-based measures and their elements are accessible for different types of cloud service users, it can be said that they can be measured by monitoring tools or monitoring methods by users in cloud environments.

Table 6-1. Time behavior measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Mean response time	$m_1 = \sum_{i=1}^{n} \frac{T_i}{n}$ (1) T_i: Response time to a user task or system task at i-th evaluation. n: Total number of response evaluations	Doable	All	All
Response time adequacy	$m_2 = T/t$ T: Mean response time evaluated from (1) t: The specified target response time	Doable	All	All
Mean turnaround time	$m_3 = \sum_{i=1}^{n}(T_i - C_i)/n$ (2) T_i: Starting time of a job i C_i: The completion time of the job i n: Number of evaluations	Doable	All	All
Turnaround time adequacy	$m_4 = T/t$ T: Mean turnaround time evaluated by (2) t: The specified target turnaround time	Doable	All	All
Mean throughput	$m_5 = \sum_{i=1}^{n}(\frac{J_i}{O_i})/n$ J_i: The number of completed jobs during the i – th observation time O_i: i-th observation time period n: The number of observations	Doable	All	All

6.1.2 Resource utilization

Resource utilization is defined in ISO/IEC 25010 [1] as :

> «degree to which the amounts and types of resources used by a product or system, when performing its functions, meet requirements. »

Thus, as explained in table 6-2 the measures of this sub-characteristic are relatively hardware-based. Since resource utilization is the core part of IaaS and regarding the adapted perspective (figure 6-2) and the explanations in chapter 6, IaaS is the subset of PaaS and SaaS, so, it can be said that resource utilization is an intrinsic characteristic of the three cloud service models (IaaS, PaaS and SaaS).

Moreover, in cloud computing the term workload refers to the IT resources utilization on which an application is hosted [153]. Workload is as a result of accessing the application or jobs by users [153]. Workloads are imminent in various forms that are proportional to the type of IT resource for which it is measured [153] :

> «Servers may experience processing load, storage offerings may be assigned larger or smaller amounts of data to store or may have to handle queries on that data. Communication IT resources, such as networking hardware or messaging systems may experience different data or message traffic. In scope of the abstract workload patterns, we merely assume this utilization to be measurable in some form. »

There are five different workload patterns that IT resources can experience in cloud environments regardless of the type of cloud service models (IaaS, PaaS and SaaS) which is used (as different workload patterns can occur in all types of cloud service models) [153] :

- Static workload: equal utilization of IT resources over time;
- Periodic workload: IT resources with having a peaking utilization at reoccurring time intervals;
- Once in a lifetime workload: equal utilization of IT resources over time distributed by a strong peak which can occur only once;
- Unpredictable workload: random and unforeseeable utilization of IT resources over time;

- Continuously changing workload: the utilization of IT resources grows or shrinks constantly over time.

According to Garraghan et al. in [206], resource utilization at the server level can be correlated significantly with the workload due to some reasons [206] :

- A load balancing technique can be used in cloud environments with the aim of keeping utilization of servers in stable conditions regardless of the cloud workload environment behavior;
- There are some types of architecture that can be assigned tasks for possessing specified conditions which can only be achieved by a particular type of architecture;
- The diverse utilization rates that is higher for CPU compared with Memory or Disk, can impact the correlation between server utilization and workload. Since a few tasks produce high utilization rates and require particular server characteristics.

Further, as described in [206] the wasted utilization of resources as a result of task terminations in cloud environments (IaaS, PaaS and SaaS) also can impact on workload. In addition, dynamicity of cloud environments can have effects on resource utilization. Measuring this dynamicity can provide a greater insight into the nature of cloud computing [206]. However, due to confidentiality concerns, there is a lack of such data related to the cloud operational environments in reality to analyze [206].

Regarding the nature of these measures, measuring can be feasible by having direct access to the physical resources in cloud environments. Obviously in terms of SaaS and PaaS, cloud users cannot have access to the physical resources (figure 5-3 in chapter 5).

In addition, in terms of IaaS the cloud users in fact cannot have direct access to the shared physical resources among various tenants in cloud environments based on security rules. Because these available resources for cloud users in IaaS, are virtual and regarding the hardware virtualization in previous chapter section 5.1, the users of IaaS services in fact have

access to the virtual hardware through virtualization techniques defined by cloud service providers (the owner of the cloud service) (figure 5-3 in chapter 5). Therefore, measuring the presented measures in table 6-2 for individual and organization may not be feasible.

On the other hand, cloud service providers are responsible to provide necessary resources based on the users' requirements and by the help of various techniques such as virtualization and load balancing can ensure the adequacy of provided resources. Regarding the measures in table 6-2 which are related to the set of users' tasks, if a cloud provider is able to distinguish the various tasks of cloud users, it would be a security violation as from users' perspective, data must be confidential even for the cloud providers. Accordingly, these measures are not measurable for cloud service providers as well.

Table 6-2. Resource utilization measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Mean processor utilization	$m_6 = \sum_{i=1}^{n}(\frac{J_i}{O_i})/n$ J_i: Processor time used to execute a set of jobs in i-th observation O_i: Operation time to perform the jobs in i-th observation n: The number of observations	Undoable	All	-
Mean memory utilization	$m_7 = \sum_{i=1}^{n}(\frac{J_i}{O_i})/n$ J_i: Memory size used to execute a set of jobs for i-th sample processing O_i: Available memory size to perform the jobs in i-th sample processing n: The number of sample processed	Undoable	All	-
Mean I/O device utilization	$m_8 = \sum_{i=1}^{n}(\frac{J_i}{O_i})/n$ J_i: The I/O device(s) duration busy time to perform a set of jobs for i-th observation O_i: Available memory size to perform the jobs in i-th sample processing n: The number of sample processed	Undoable	All	-
Bandwidth utilization	$m_9 = B/b$	Undoable	All	-

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
	B: Bandwidth of actual transmission evaluated during the time of a set of jobs b: The capacity of available bandwidth to perform a set of jobs			

6.1.3 Capacity

The cloud capacity refers to the total amount of resources owned and maintained by cloud service providers and requires proper planning since it impacts on the cost of infrastructure and the ability to meet the requirements [207]. Thus, this is the cloud service provider who can control the cloud capacity and it may not be transparent for the users. Capacity measures are considered to evaluate the degree to which the maximum limits of a cloud service accommodate the demands [160]. The maximum limit is specified by a goal value that may theoretically be far from possible realistic value [160]. However, since this is a part of cloud configurations which is fulfilled by cloud providers, it can be considered that the measures of capacity (table 6-3) can be measured mostly by cloud service providers themselves. Except "transaction/job processing capacity" measure that can be measured also by individual and organization with monitoring methods given that the user would be able to count the number of completed transactions or jobs during observation time in cloud environments. As an example we can consider Gmail which is a cloud service proposed by Google. A Gmail user would not be able to precisely specify the number of other users who can simultaneously access Gmail or the number of users added in observation time. But counting the number of the completed jobs for users can be more practical.

Table 6-3. Capacity measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Transaction/job processing capacity	$m_{10} = T/t$ T: The number of completed transactions/jobs in observation time t: Observation time	Doable	All	All
User access capacity	$m_{11} = \sum_{i=1}^{n} \frac{T_i}{n}$ T_i: The maximum number of users who can simultaneously access the system at i-th observation n: The number of observations	Doable	All	Cloud Service Provider
User access increase adequacy	$m_{12} = U/t$ U: The number of users added in observation time t: Observation time	Doable	All	Cloud Service Provider

6.1.4 Elasticity

Elasticity can be considered as the system capability to increase or decrease the resources dynamically or in a on-demand way [208]. In cloud computing, there are two concepts related to the elasticity which are fundamental to benefit from the cloud pay-per-use property of IaaS and PaaS [153]:

- Scaling-out (or horizontal scaling): increasing the number of independent resources e.g. servers, when an application needs more processing power, storage, etc.;
- Scaling-up (or vertical scaling): improving an application performance by developing resources capabilities on which it runs without modifying their number.

Meanwhile, it should be noticed that vertical or horizontal scaling of the resources can cause the deviation of the current number of allocated resources from the actual resource demand [209]. Since the elasticity systems accuracy is different in each cloud system, thus it can directly effect on performance efficiency.

> «The customer requires the software, platform or infrastructure to behave in an elastic manner, providing exactly the number of resources required. The provider, on the other hand, seeks to balance the load of the different applications on a platform or infrastructure or the different tenants of a software, so that the whole system more or less experiences static workload that best utilizes the resources assigned to the customer(s) by the provider [153]. »

Subsequently, regarding this fact and the explanations in ISO/IEC 17788 [24] and the discussions in chapter 4, in a normal situation the paucity of the resources would not be tangible for the users in cloud environments. Therefore, assessing the measures of elasticity would be only practical for the cloud service providers (the owner of the cloud services).

Table 6-4. Elasticity measures

Measures (adapted from ISO/IEC 19086-1) [150]	**Evaluation Function (adapted from ISO/IEC 19086-1) [150]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Speed	m_{13} = (R+C)/n R: The time lasted to resource re-allocation C: The time lasted to change work load n: The total number of requests	Doable	All	Cloud Service Provider
Provision	$m_{14} = R - N$ R: The number of actually allocated resources N: Total number of needed resources	Doable	All	Cloud Service Provider

6.2 The verification of security measures

The proposed security measures are to evaluate the degree to which a cloud system can protect data and information (ISO/IEC 25023 [160]). To assess the security measures it is recommended to simulate an attack since during the usual test, a security attack does not occur [160].

6.2.1 Confidentiality

Confidentiality refers to the rules and restrictions which create limitations on access to the specified types of information and protect users' data not to be attainable even by the cloud service providers [210].

> « Confidentiality is often achieved by encryption. However, when a company's data is stored in a Cloud environment, one has to consider the problem of long-term confidentiality, meaning that past and present encryption schemes are expected to be insecure in a long run (e.g. 30 years). Moreover, if one is going to process data in the Cloud, this data will usually be decrypted which also poses threat to Confidentiality [211]. »

According to the measures which are presented in table 6-5, since the cryptographic algorithms used for confidentiality would be confidential information of cloud service providers, thus the measures of this sub-characteristic presented in the table 6-5, would be only measurable for cloud service providers (the owner of the cloud services), except «access controllability » that may be measurable (by testing) for all types of users. Since data is the precious asset of users, it is believed that users are able to specify the number of data items that require access control. In addition, because of multi-tenant cloud services, confidentiality would be crucial for all three types of cloud service models (IaaS, PaaS, SaaS).

Table 6-5. Confidentiality measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Access controllability	$m_1 = 1 - D/d$ D: The number of confidential data items that can be accessed without authorization d: The number of data items that require access control	Doable	All	All
Data encryption correctness	$m_2 = D/d$ D: The number of data items encrypted or decrypted correctly	Doable	All	Cloud Service Provider

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
	d: The number of data items that require encryption or decryption			
Strength of cryptographic algorithms	$m_3 = 1 - A/a$ A: The number of cryptographic algorithm broken or unacceptably risky in use a: The number of used cryptographic algorithms	Doable	All	Cloud Service Provider

6.2.2 Integrity

To assess the degree to which a system or component prevents unauthorized access to, or modification of data and computer programs, integrity measures are introduced (table 7-6) [160]. Integrity assures that data is trustworthy and accurate [210].

Data integrity in cloud systems preserves information integrity and prevent data to be lost or modified by unauthorized users [212]. Data integrity is essential to provide SaaS, PaaS and IaaS services [212]. To assess data integrity, the number of corrupted data can be achievable by log checking and testing the cloud services. In this case, individual, organization and cloud service provider can assess this measure regarding their own data.

Internal data corruption prevention refers to the implemented available prevention methods for data corruption [160]. As it is explained in ISO/IEC 25023 [160], some examples for data corruption prevention are data backup, comparing data to the reference data and storing data in multiple mirror sites.

A (data) buffer is a part of physical memory storage to temporarily store data. Buffers are helpful to improve performance in cloud environments. In case of existing more data than size of a buffer, buffer overflow occurs and causes data corruptions. Buffer overflow is a coding mistake and normally is due to insufficient bounds checking [160, 213]. When a buffer

overflow happens, an attacker could exploit and have access to the data and the system. For example, buffer overflow can cause a hypervisor attack exploits this vulnerability [214].

> « In cloud computing, a hypervisor abstracts the hardware of a shared physical server into virtualized hardware. On this virtual hardware, different operating systems and middleware are installed to host applications sharing the physical server while being isolated from each other regarding the use of physical hardware, such as central processing units (CPU), memory, disk storage, and networking [153]. »

However, buffer overflow as explained in table 6-6, cannot be measurable in cloud computing.

Table 6-6. Integrity measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Data integrity	$m_4 = 1 – D/d$ D: the number of corrupted data by unauthorized access d: The number of data items for which data corruption or modification have to be prevented	Doable	All	All
Internal data corruption prevention	$m_5 = D/d$ D: The number of implemented data corruption prevention methods d: The number of available and recommended data corruption prevention methods	Doable	All	Cloud Service Provider
Buffer overflow prevention	$m_6 = M/m$ M: The number of memory accesses with user input that are bounds checked m: The number of memory accesses with user input in software modules	Undoable	All	-

6.2.3 Non-repudiation

A typical example of disputes that could occur in cloud environments is transferring a message from Alice to Bob, where Alice claims sending a message to Bob, but Bob denies receiving it, or Bob claims he receives a message from Alice, but Alice denies sending it to Bob [215]. Another prevalent example is that an authorized customer but with malicious intentions alters the stored data in cloud but denies it [215]. Non-repudiation by creating, collecting, validating and maintaining cryptographic evidence can support the settlement of those disputes [215, 216]. Non-repudiation assures that the authorized user cannot deny the performed modifications by him/her [217]. There are three types of non-repudiation services that need to be provided by the originator, the delivery agent(s) and the recipient(s) [215] :

> - «Non-Repudiation of Origin (NRO) provides the recipient(s) of a message with the proof of the origin of the message. It will protect against any attempt by the originator to falsely deny sending the message;
> - Non-Repudiation of Delivery (NRD) provides the originator of a message with the proof that the message has been delivered to the originally specified recipient(s);
> - Non-Repudiation of Receipt (NRR) provides the originator of a message with the proof of receipt of the message. It will protect against any attempt by the recipient(s) to falsely deny receiving the message. »

In addition, there are four phases incorporate with the non-repudiation services [215, 218] :

- Evidence generation- the evidences are generated by the originator, the recipient, the delivery agent, or the trusted third party, depending on concrete designs and applications;
- Evidence transfer- in this phase, fairness, efficiency and timeliness which are required in the communications can be provided;
- Evidence verification and storage- the newly received evidence needs to be verified to achieve confidence to ensure the supplied evidence will indeed be acceptable in the event of a dispute. Valid evidence needs to be stored safely;

- Dispute resolution- in case of occurring a dispute, an adjudicator will be invoked to settle the dispute regarding the provided non-repudiation evidence by the dispute parties.

Regarding table 6-7, non-repudiation can be measured by digital signature usage (ISO/IEC 25023 [160]). In public key cryptography, a user has a pair of cryptographic keys [215] : a private key (protected secretly by the owner) and a public key that can be published. One of the most important public key cryptography applications is digital signature, which is defined in [215] as :

> « A signature scheme Σ is made up of three essential algorithms: KeyGen, Sign and Verify.
> Given a system parameter param, these algorithms work as follows.
> KeyGen(param)→(sk , pk). On input param, this algorithm generates a private-public key pair (sk, pk). The public key includes the description of the message space M and signature space S.
> Sign(param,m, sk)→σ. On input param, a message m ∈ M and a private key sk, this algorithm generates a signature σ ∈ S.
> Verify(param, m ,σ, pk)→ 0,1. On input param, a message-signature pair (m , σ) ∈ M ×S and a public key pk, this algorithm verifies the validity of the signature and outputs a decision "1" or "0". (m ,σ, pk) is said to be a valid triple if Verify outputs "1".
> Consistence. For any message m ∈ M and any key pair (sk, pk) generated by KeyGen, Verify (param,m, Sign(param,m, sk),pk) = 1. »

Subsequently, we believe that using digital signature is a part of security policy of cloud service providers which is confidential. Thus, measuring non-repudiation with digital signature, at this stage, with the represented measure in table 6-7 which is extracted from ISO/IEC 25023 [160], would be practical only for cloud service providers (the owners of the cloud services).

Table 6-7. Non-repudiation measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Digital signature usage	$m_7 = E/e$ E: The number of events that ensure non-repudiation using digital signature e: The number of events which require non-repudiation by using digital signature	Doable	All	Cloud Service Provider

6.2.4 Accountability

Accountability refers to how the information and data are controlled, the remedies of any failure and clarifications of any action [219-221]. As table 6-8 presents, the measures of this sub-characteristic are to evaluate the degree to which the actions of an entity can be traced [160]. As it is explained in ISO/IEC 25023 [160], these measures can be assessed regarding the given measurement functions based on log policy.

Table 6-8. Accountability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
User audit trail completeness	$m_8 = A/a$ A: The number of accesses recorded in all logs a: The number of accesses to tested system or data	Doable	All	All
System log retention	$m_9 = T/t$ T: Duration for which the system log is retained in stable storage t: Specified retention period to keep the system log in stable storage	Doable	All	All

6.2.5 Authenticity

Authenticity is about the process of approving user's identity before having access to different types of provided cloud services [222]. It is used to make sure whether the user or the application is qualified to access or claim [223]. The most common authentication mechanisms in network environments are log-on credentials, multifactor authentication, third party authentication, simple text passwords, 3D password objects, graphical passwords, biometric authentication and digital device authentication [160, 223]. In cloud computing, a single mechanism or combination of these authentication mechanisms can be implemented [223]. The table 6-9 describes the measures of authenticity which are measurable and crucial for three types of cloud service models (IaaS, PaaS, SaaS). Cloud users based on the specified authentication rules and mechanisms in the system and comparing with the agreed service level agreement (SLA) or in the general documents related to the cloud service providers would be able to assess this sub-characteristic.

Table 6-9. Authenticity measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Authentication mechanism sufficiency	$m_{10} = A/a$ A: The number of provided authentication mechanisms a: The number of specified authentication mechanisms	Doable	All	All
Authentication rules conformity	$m_{11} = R/r$ R: The number of implemented authentication rules r: The number of specified authentication rules	Doable	All	All

6.2.6 Traceability

In ISO/IEC 25011 [148], traceability is defined as the degree to which the outcomes of cloud services can be traced to or from the user needs; “from the user’s needs” is for example the customer of the online-order room needs to know the reservation progress, and “to the user’s needs” is for example the hotel needs to know about the payment progress of the customer [148]. This measure (the table 6-10) is essential for IaaS, PaaS and SaaS to meet the related requirements.

Table 6-10. Traceability measures

Measures (adapted from ISO/IEC 25011) [148]	**Evaluation Function (adapted from ISO/IEC 25011) [148]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Traceable outcomes	m_{12} = O/o O: The number of traceable outcomes to/from the user needs o: The total number of outcomes to/from the user needs	Doable	All	All

6.3 The verification of reliability measures

Reliability represents the possibility that a cloud service is operational during a specified time without any failure [224].

> « Internal reliability measures are used for predicting if the completed system/software product in question will satisfy prescribed reliability needs during the development of the system/software product. External reliability quality measures are used to assess attributes related to the behaviour of the system of which the software is a part during execution testing to indicate the extent of reliability of the software in that system during operation. Systems and software are not distinguished from each other in most cases (ISO/IEC 25023 [160]). »

6.3.1 Maturity

Maturity measures are to evaluate the degree to which a system or component can meet the requirements of reliability under normal operation [160]. As the table 6-11 presents, fault correction, mean time between failure and failure rate are the first three measures of maturity. To analyze these measures, it is necessary to clarify the definitions of failures, faults and errors. Failures in a system are as a result of occurring errors which are due to faults [225] :

- Faults: It is the incapability of a system in performing the essential/required task which is as a result of the abnormal state or bug in some parts of a system [225-230].
  Some of the important types of faults are [225, 226, 228, 229] :

  - Network fault: cloud computing resources can be accessed over a network (Internet); thus it can be said that the network fault is the predominant cause of failures in cloud environments. Partitions in the network, packet loss, congestion, failure of the destination node are some example of network faults;
  - Physical faults: These faults occur mostly in hardware resources (i.e. faults in CPU, in memory, in storage, etc.);
  - Process faults: This type of fault might occur due to shortage of resources, bugs in software, inefficient processing capabilities, etc.;
  - Service expiry faults: It occurs when the service time of a resource expires while it is in the usage of an application that leased it.

- Errors: They occur mainly as a result of the presence of faults [225]. The performance of a system component erroneously can cause partial or in some cases complete failure in the system [225, 226, 230]. In cloud computing different types of errors occur [225] :

  - Network: packet corruption- packet loss- network congestion;
  - Software: memory leak- numerical exception;
  - Miscellaneous: permanent errors- intermittent errors- transient errors.

- Failure: It refers to the misbehavior of a system which is experienced by a user (a human or some other computer systems) [225]. A failure can be recognized only when the output or outcome of a system is incorrect [225, 226, 230, 231]. Various failures can be categorized in different groups [225] :

  - Omission: receive omission- send omission;
  - Hardware: machine failure- disk failure- memory failure- CPU failure- device failure;
  - Software: OS failure- Application failure- user defined exception- unhandled exception- unexpected input;
  - Network: site failure- link failure- configuration change- device failure;
  - Response: value failure- state transition failure;
  - Miscellaneous: timing failure- arbitrary failure.

In addition, since testing a system is a part of internal policy of a cloud provider, measuring test coverage would not be practical for individuals and organizations.Therefore, we can say that fault correction and test coverage can be measured by cloud service providers and failure rate and mean time between failure are measurable for individuals as well as organizations.

Table 6-11. Maturity measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Fault correction	$m_1 = F/f$ F: The number of corrected reliability-related faults f: The number of detected reliability-related faults	Doable	All	Cloud Service Provider
Mean time between failure	$m_2 = O/f$ O: Operation time f: The number of occurred failures	Doable	All	All
Failure rate	$m_3 = F/t$ F: The number of detected failures in observation time t: Observation time	Doable	All	All

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Test coverage	$m_4 = F/f$ F: Number of performed system or software functions f: Number of system or software functions included in their associated test suits	Doable	All	Cloud Service Provider

6.3.2 Availability

Availability measures are used to ensure that a system would be practical and accessible when it is required [160]. According to table 6-12, in which the measures of this sub-characteristic are presented, system availability is one of its measures. It refers to the duration of operational time in which the system is actually available [160]. This time can be monitored during regular days and extended to especial days such as holidays or weekend [160].

Mean down time is the second measure and refers to the duration of time that system stays unavailable in case of existing a failure [160]. According to the definition of failure in the previous section, it can be measurable.

Table 6-12. Availability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
System availability	$m_5 = T/t$ T: Provided system operation time t: Specified system operation time	Doable	All	All
Mean down time	$m_6 = T/b$ T: Total down time b: The number of observed breakdowns	Doable	All	All

6.3.3 Fault Tolerance

Fault tolerance in cloud computing alludes to a situation in which a fault occurred and the cloud service is able to detect and recover it without any detrimental damage to the final outcome of the cloud service [227]. To detect and handle the faults that they may occur due to hardware failure or software faults, fault tolerance approaches are necessary [225]. Since faults and failure can occur in three service models of cloud computing (IaaS, PaaS, SaaS), thus fault tolerance is crucial to all these service models.

In table 6-13, the measure of failure avoidance refers to the ability of the system to control the fault patterns to avoid critical failures [160]. As explained in ISO/IEC 25023 [160], this can be measured during test of the system by all types of cloud user.

Redundancy of components is about installation of system components redundantly to avoid system failure [160]. As an example, in safety-critical systems, some parts of the control system could be duplicated to increase reliability of the system [160]. Since the installation of the system component is the internal policy of the provider, therefore, individual and organization cannot measure it.

Mean fault notification time refers to the ability of the system to report the occurrence of faults immediately. Since an end user would not identify the occurred faults in the system, thus, this measure cannot be evaluated by individuals and organizations.

Table 6-13. Fault tolerance measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Failure avoidance	$m_7 = F/f$ F: The number of avoided critical failure (based on test cases) f: The number of executed test cases of fault pattern during testing	Doable	All	All

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Redundancy of components	$m_8 = C/c$ C: Number of system components redundantly installed c: Number of system components	Doable	All	Cloud Service Provider
Mean fault notification time	$m_9 = \sum_{i=1}^{n}(T_i - C_i)/n$ T_i: The time at which the fault i is reported by the system C_i: The time at which fault i is detected n: Number of faults detected	Doable	All	Cloud Service Provider

6.3.4 Recoverability

Regarding the definition of recoverability in ISO/IEC 25010 [1], which is the ability of a system to recover data directly affected and re-establish the desired state of the system in case of an interruption or a failure, this sub-characteristic would be crucial for all three cloud service models (IaaS, PaaS, SaaS). In some cases, following a failure, the system will be down for a while. It is important that the duration of this time be as short as possible.

In terms of backup data completeness, it can be said that backup is an internal policy of the cloud service provider and the cloud provider is responsible for the safety of the data. Thus, individual and organization would not be able to measure that.

Subsequently, since a cloud end user and an organization can experience the failure of the system, thus measuring down time of the system would be easy.

Table 6-14. Recoverability measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Mean recovery time	$m_{10} = \sum_{i=1}^{n} T_i / n$ T_i: Total time to recover the downed software/system for each failure i n: Number of failures	Doable	All	All
Backup data completeness	m_{11} = D/d D: The number of backed up data items d: The total number of data which require backup	Doable	All	Cloud Service Provider

6.3.5 Continuity

Continuity is defined in ISO/IEC 25011 [148] as :

> « degree to which the IT service is provided under all foreseeable circumstances, including mitigating the risks resulting from interruption to an acceptable level. »

When a customer decides to benefit from provided cloud services, it is expected that the risks and the consequences resulting from an interruption or failure are supported by cloud service providers. Thus, the cloud service provider is supposed to predict all risky situations in order to provide the solutions and equipment. In this way, the system can relatively be guaranteed to perform specified functions under specified conditions to provide consistent and stable outcomes; which is the definition of reliability in ISO/IEC 25010 [1] and the definition of IT service reliability in ISO/IEC 25011 [148]. In chapter 5, the measure of continuity is defined as supported cloud services. But since, there is not any defined measure for this sub-characteristic in related standards, we propose “customer support policy” as its measure, because supporting cloud services will be performed through customer support policy of each cloud service provider. In addition, in some cases to be able to continue using the cloud service, some features are required. These features should be treated according to the policy of

supporting customers of the cloud service provider. Therefore, assessing this measure for cloud users would be important and practical.

To design the measurement method for the proposed measure of «customer support policy », we adapt the precepts defined in ISO/IEC 25021 [232], since the purpose of this standard is to design a set of quality measure elements (QME) to be used in the product life cycle for the purpose of Systems and Software Quality Requirements and Evaluation (SQuaRE). Its proposed steps are discussed as follows :

Step1) Identifying the quality measure element (QME) and objectives.
QME is defined in this standard [232] as :

> «measure defined in terms of a property and the measurement method for quantifying it, including optionally the transformation by a mathematical function. »

Step 2) Identifying the property for quantifying related to QME.
Step 3) Defining the property and sub properties
Step 4) Designing the model of the properties to be quantified
Step 5) Assigning the unit of measurement (formula) and scale type

In this case, regarding the proposed measure, the information items to design a QME are provided as follows (adapted from ISO/IEC 25021 [232]) :

- QME name - Customer support policy;
- Target entity – Foreseeable circumstances of the system, bugs/errors in a specific time, required features;
- Objectives and property to quantify – The objective is to measure the quality of the customer support policy of the cloud service providers to be able to continue the usage of the cloud services with reference to definition of continuity in ISO/IEC 25011 [148];

- The properties that need to be measured are : 1- The number of supported foreseeable circumstances to mitigate the risks resulting from interruption to an acceptable level, 2- The total number of foreseeable circumstances, 3- Number of fixed bugs or errors in a specific time, 4- The total number of informed bugs in a specific time, 5- The number of added new features in a specific time, 6- Total number of required features in a specific time;
- Relevant quality measure(s) – Reliability (characteristic level) defined in ISO/IEC 25010 [1] and continuity (sub-characteristic level) defined in ISO/IEC 25011 [148];
- Measurement method – Based on users' requirements the number of supported foreseeable circumstances to mitigate the risks resulting from interruption to an acceptable level, number of fixed bugs or errors in a specific time, the number of added new features, in a specified duration of time can be counted. Then, dividing them by the total number of foreseeable circumstances, the total number of informed bugs/errors in a specific time, total number of required features in a specific time respectively, and adding the results;
- Input for the QME – Cloud users' requirements in supporting foreseeable circumstances, fixing bugs and errors and adding required features;
- Unit of measurement for the QME - The number of foreseeable circumstances specified in cloud service providers' policy, the number identified bugs/errors in a specific time and the number of required features in a specific time;
- Numerical rules – Count the number of foreseeable circumstances specified in cloud service providers' policy, the number identified bugs/errors in a specific time and the number of required features in a specific time and dividing them by the total number of foreseeable circumstances, total number of identified bugs or errors and required features respectively. Finally, adding the results to produce a unique metric;
- Scale type – ratio;
- Context of QME – This QME is usable to measure the reliability and continuity of the cloud services;
- System life cycle process(es) – Testing, implementation, operation and maintenance.

Regarding the aforementioned explanations, table 6-15 presents the proposed measures for the sub-characteristic of continuity.

Table 6-15. Continuity measures

Measure	Evaluation Function (partially adapted from ISO/IEC 25011) [148]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Customer support policy	m_{12} = S/s + B/b + F/f S: The number of supported foreseeable circumstances to mitigate the risks resulting from interruption to an acceptable level s: The total number of foreseeable circumstances B: Number of fixed bugs/errors in a specific time b: The total number of informed bugs in a specific time F: The number of added new features in a specific time f: Total number of required features in a specific time	Doable	All	All

6.4 The verification of freedom from risk measures

As explained in ISO/IEC 25010, stakeholders comprise three types of user :

- Primary user who has interactions with the system to achieve the primary goals;
- Secondary user who acts as a supporter (e.g. content provider and system manager);
- Indirect user who receives output, however does not have any interaction with the system.

According to the definition of freedom from risks in ISO/IEC 25010 [1], risks of undesirable consequences can effect the following types of stakeholders [173] :

- User of a system: health and safety while using the system – detrimental consequences of failing to achieve the goal outcome;

- Organization using a system: organization's reputation loss or finances damage from errors as a result of the system usage – risks of inadequate operational safety;
- Organization developing a system or product: economic consequences risk if the developed target system or product does not have the intended quality – economic and reputational consequences risks when a system or product not being purchased or used as a result of quality deficiency;
- Wider community: health and safety consequences risks.

6.4.1 Economic risk mitigation

To assess the impact of cloud services quality on economic goals related to financial status, efficient operation, commercial property, reputation, or the other resources that could be at risk, the measures of economic risk mitigation can be helpful [173]. These measures are used to mitigate the risk of inadmissible economic outcomes, based on actual values from historical data [173]. Financial risks are part of economic risks. Financial risks occur when a project could not meet benefits [233]. This is essential in cloud computing, since cost benefit is one of the appealing reasons to switch to cloud. In addition, by assessing the measures of economic risk mitigation (table 6-16), cloud service users can gain a broader view to evaluate the cost of proposed cloud services.

Table 6-16. Economic risk mitigation measures

Measures (adapted from ISO/IEC 25022) [173]	**Evaluation Function (adapted from ISO/IEC 25022) [173]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Return on investment (ROI)	$m_1 = (B – I)/I$ B: Additional obtained benefits I: Invested amount	Doable	All	Organization and Cloud Service Provider
Time to achieve return on investment	$m_2 = T$ T: Time to achieve ROI	Doable	All	Organization and Cloud Service Provider
Business performance	$m_3 = Aa/At$	Doable	All	Organization and Cloud

Measures (adapted from ISO/IEC 25022) [173]	Evaluation Function (adapted from ISO/IEC 25022) [173]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
	A: Profitability or sales of the company (a: actual, t: target)			Service Provider
Benefits of IT investment	m_4 = Ba/Bt B: The benefit of IT investment (a: actual, t: target)	Doable	All	Organization and Cloud Service Provider
Service to customers	m_5 = S/s S: Actual level of service s: Intended level of service e.g: the average waiting time to obtain customer service	Doable	All	All
Loyal cloud customers to a specific cloud service provider	m_6 = L/C L: The number of the loyal customers in a specified time C: The total number of the customers in a specified time	Doable	All	Organization and Cloud Service Provider
Revenue from each customer	m_7 = R R: Revenue from a customer	Doable	All	Organization and Cloud Service Provider
Errors with economic consequences	m_8 = E/U E: The number of errors with economic consequences (e.g: data corruption) U: The total number of usage situations (transactions or time)	Doable	All	Organization and Cloud Service Provider

6.4.2 Health and safety risk mitigation

Each product, system or service may contain some deficiencies. The effects of these deficiencies are not always in functional aspects of the systems. But they might rather be unbearable and cause some illnesses for the users such as fatigue, headaches and stress. The first two measures of this sub-characteristic (illustrated in table 6-17) are related to these issues. On the other hand, a proper example to explain the importance of the last measure of this sub-characteristic (Safety of people affected by use of the system) can be cloud systems in healthcare domains. In the work which is done in [234], the patient's real-time vital diseases

symptoms are collected by wireless body area network (WBAN) and analyzed with the patient's historical repository of diseases, habits, rehabilitations and genetics in healthcare cloud platform. Safety of people can be affected by use of this cloud platform. Since, in some cases, the collected data from patients could be different from the reality. Thus, some patients could be prescribed treatment incorrectly [173].

Table 6-17. Health and safety risk mitigation measures

Measures (adapted from ISO/IEC 25022) [173]	**Evaluation Function (adapted from ISO/IEC 25022) [173]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
User health reporting frequency	$m_9 = U/u$ U: Number of users reporting health problems u: Total number of users Ex: fatigue, headaches, etc.	Doable	All	Organization and Cloud Service Provider
User health and safety impact	$m_{10} = \frac{1}{T_b}\sum_{i=1}^{n}(T_{a_i} \times S_i)$ n: Number of affected people Ta_i: Length of time for which the i-th person is effected S_i: Degree of significance of the impact on the i-th person T_b: Length of time from start of system in operation Ex: The stress as a result of the difficulty of using a system with a poor user interface	Doable	All	Organization and Cloud Service Provider
Safety of people affected by use of the system	$m_{11} = P/T$ P: The number of people put at hazard T: Total number of people who could be affected by use of the system	Doable	All	Organization and Cloud Service Provider

6.4.3 Environmental risk mitigation

Environmental risk mitigation measures are used to evaluate the environmental impact by using the system [173]. As already explained in chapter 4, the measure of environmetal risk mitigation in ISO/IEC 25022 [173] is defined as « environmental impact ». However, a number of studies have postulated that the major issue associated with the cloud computing technology

related to the environmental impact would be energy efficiency and in particular reducing power consumption [235, 236].

The literature in the domain of energy efficiency in computer science shows there are various methods in cloud computing to reduce power consumption [237-239]. The main reason lies in the operational budget and the environmental issues [237] such as CO2 emission. Since lower power consumption can result in lower CO2 emission [238]. Thus, it is worth to evaluate this measure in order to hamper extra expenditures in power consumption.

As an example, according to an article released by Yahoo [240], despite the cloud-based notion of internet, it depends on more than millions of physical servers in data centers around the world, connected with undersea cables, switches and routers which a lot of energy required to run. A significant amount of that energy produces from power sources which emit CO2 into the air as they burn fossil fuels [240]. Based on this article [240] :

> « One estimate from British environmental consultancy Carbonfootprint puts it between 1 kg and 10 kg of CO2 per Google search.»

Accordingly, we propose power consumption as the measure of environmental risk mitigation in cloud computing. Regarding the explanations in section 6.3.5 (continuity), to design the measurement method for the proposed measure of «power consumption», we adapt the precepts defined in ISO/IEC 25021 [232]. Thus, the information items to design a QME are provided as follows (adapted from ISO/IEC 25021 [232]) :

- QME name – Power consumption;
- Target entity – Power;
- Objectives and property to quantify – To know the power consumption. Regarding the operational budget and environmental concerns, power consumption can affect the quality characteristic of freedom from risk defined in ISO/IEC 25010 [1];

- The properties that need to be measured are : 1- The actual power consumption, 2- The target power consumption;
- Relevant quality measure(s) – Freedom from risk (characteristic level) defined in ISO/IEC 25010 [1] and environmental risk mitigation (sub-characteristic level) defined in ISO/IEC 25022 [173];
- Measurement method – The actual power consumption is divided by the target power consumption to estimate the current power consumption;
- Input for the QME – The power consumption in a specified duration of time;
- Unit of measurement for the QME - The power consumption;
- Numerical rules – Dividing;
- Scale type – Ratio;
- Context of QME – This QME is usable to measure the environmental risk mitigation of the cloud services;
- System life cycle process(es) – Operation and maintenance.

Regarding the aforementioned explanations, the table 6-18 presents the proposed measures for the sub-characteristic of environmental risk mitigation. It should be noted that since the power consumption may be part of confidential information related to the cloud service providers, therefore, this measure would be measurable only for cloud service providers (the owner of the cloud service). In addition, regarding to the explained perspective (figure 6-2), the proposed measure would be applicable for three cloud service models.

Table 6-18. Environmental risk mitigation measures

Measure	**Evaluation Function**	**Type of category** (Doable/ Undoable)	**Applicable for** (SaaS/PaaS/ IaaS)	**Type of user** (Individual/ Organization/ Cloud Service Provider)
Power consumption	m_{12} = Aa/At A = Power consumption (a: actual, t:target)	Doable	All	Cloud Service Provider

6.5 The verification of usability measures

Usability is a quality characteristic that can assess simplicity of any service or product, fulfill users' requirements and users' satisfaction [241]. As explained in ISO/IEC 25023 [160] :

> «Internal and external measures for usability make comparisons between stated design conventions, specific guidelines or specification for usability and actually developed documented design, prototype or executable system/software. Therefore, it is very important to elicit end user's requirements and create well specific specification for usability by considering characteristics and measures of quality in use as well as user centred design concept and human ergonomics view. For example, usability specific guideline, templates, or check list are necessary to explain in detail what kinds of messages are easy to understand for end users. »

In addition, the results of assessing usability measures would be subjective [160]. Thus, an ordinal scale can be considered in case of having difficulties such as 10 for excellent and 1 for bad.

6.5.1 Appropriateness recognisability

As stated in the literature [242-244] and ISO/IEC 25023 [160], cloud users need to be able to select a cloud service which can meet their needs for their intended use. The measures of appropriateness recognisability (table 6-19) are helpful to assess the degree to which cloud users are able to recognize a proper cloud service for their purposes and understand how it can be used [160]. Therefore, these measures are applicable for all three cloud service models (IaaS, PaaS, SaaS).

Table 6-19. Appropriateness recognisability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Description completeness	$m_1 = S/s$ S: The number of usage scenarios described in the product documents s: The number of usage scenarios of the product	Doable	All	All
Demonstration coverage	$m_2 = T/t$ T: The number of tasks with demonstration features t: The number of tasks that could benefit from demonstration features	Doable	All	All
Entry point self-descriptiveness	$m_3 = S/s$ S: The number of landing pages of user console that explain the purpose of service s: The number of landing pages of user console in a service	Doable	All	All

6.5.2 Learnability

As it can be deduced from the measures of learnability in table 6-20 [160], learnability is the degree of a cloud service understandability. Thus, to benefit from cloud services measuring this sub-characteristic would be necessary for all types of cloud service models.

Table 6-20. Learnability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
User guidance completeness	$m_4 = F/f$ F: The number of functions described in user documentation f: The number of implemented functions	Doable	All	All
Entry fields defaults	$m_5 = E/e$	Doable	All	All

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
	E: The number of entry fields whose default values have been automatically filled in during operation e: The number of entry fields that could have default values			
Error message understandability	$m_6 = M/m$ M: Number of error messages which state the reason of occurrence and suggest the ways of resolution m: Number of implemented error messages	Doable	All	All
Self-explanatory user interface	$m_7 = I/i$ I: Number of information elements and steps that are presented in a way that the user could understand i: Number of information elements and steps needed to complete common tasks for a first time user	Doable	All	All

6.5.3 Operability

To evaluate the degree of simplicity of the cloud services to operate and control, the measures of operability are defined [160]. As explained in ISO/IEC 25023 [160];

> «Operability measures are expected to be measured through operational testing by representatives of operators or end users, or can be measured through static analysis such as review of requirement, design specification or user manuals. »

Table 6-21. Operability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Operational consistency	$m_8 = 1 – T/t$ T: Number of specific interactive tasks that are performed inconsistently	Doable	All	All

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
	t: Number of specific interactive tasks that need to be consistent			
Message clarity	$m_9 = M/m$ M: Number of messages that convey the right outcome or instructions to the user m: Number of messages implemented	Doable	All	All
Functional customizability	$m_{10} = F/f$ F: Number of customizable functions f: Number of functions which users could benefit from customization	Doable	All	All
User interface customizability	$m_{11} = I/i$ I: Number of customizable user interface elements i: Number of user interface elements that could benefit from customization	Doable	All	All
Monitoring capability	$m_{12} = F/f$ F: Number of functions having state monitoring capability f: Number of functions that could benefit from monitoring capability	Doable	All	All
Undo capability or re-confirmation	$m_{13} = T/t$ T: Number of tasks that provide re-confirmation or undo capability t: Number of tasks for which users could benefit from having re-confirmation or undo capability	Doable	All	All
Understandable categorization of information	$m_{14} = I/i$ I: Number of information structures familiar for users i: Number of used information structures	Doable	All	All
Appearance consistency	$m_{15} = 1 – U/u$ U: Number of user interfaces with similar items but with different appearances u: Number of user interfaces with similar items	Doable	All	All
Input device support	$m_{16} = T/t$ T: Number of tasks that can be initiated by all appropriate input modalities t: Number of tasks supported by the system	Doable	All	All

6.5.4 User error protection

Basically, there are two types of user errors [245] :

- Slips –when users have a purpose to perform an action, but end up performing another action which is often similar, this type of user error occurs. For example, inserting a “M” instead of a “N”;
- Mistakes – when users have a goal which is not proper for the current problem or task, the mistakes can be made. Even if they perform the complete steps to achieve their goals, an error will be the result of the performed steps. As an example, we can refer to a situation in which a cloud user misunderstood the meaning of a capacity warning, but thought it was a functional disability. It is also important for the subcharacteristic of learnability in section 6.5.2, in particular, for the measure of “error message understandability”.

To assess the degree to which the system is able to protect users from mistaking errors, the measures of user error protection (table 6-22) are helpful [160]. As explained in ISO/IEC 25023 [160] :

> « user error protection measures are expected to be measured through operational testing by representatives of operators or end users, or can be measured through static analysis such as review of requirement, design specification or user manuals. »

Table 6-22. User error protection measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Avoidance of user operation errors	$m_{17} = A/a$ A: Number of user actions and inputs that are protected from causing any system malfunction (for example, requesting confirmation before carrying out an action.) a: Number of user actions and inputs that could be protected from causing any system malfunction	Doable	All	All

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
User entry error correction	m_{18} = E/e E: Number of entry errors for which the system provides a suggested correct value e: Number of entry errors detected	Doable	All	All
User error recoverability	m_{19} = E/e E: Number of user errors that are designed and tested to be recovered by the system e: Number of user errors which can occur during operation	Doable	All	All

6.5.5 User interface aesthetics

In ISO/IEC 25023 [160], user interface aesthetics measures are defined as the degree to which the user interface is able to provide pleasing and satisfying interaction for users. It is believed that aesthetically pleasing interfaces can enhance user efficiency and reduce perceived interface complexity which can help to develop usability, productivity, learnability and acceptability of the system [246, 247].

Table 6-23. User interface aesthetics measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Appearance aesthetics of user interfaces	m_{20} = D/d D: Number of display interfaces aesthetically pleasing to the users in appearance d: Number of display interfaces	Doable	All	All

6.5.6 Accessibility

Accessibility measures in ISO/IEC 25023 [160] are defined as the degree to which a system can be usable for the people with different characteristics and capabilities. In cloud computing, accessibility refers to all available cloud services and computing resources over network which can be accessed from anywhere [248, 249]. As an example [249, 250] :

> «Cloud resources may be accessed from browsers running JavaScript/AJAX or at a program level using web services standards such as SOAP as a request encoding standard and HTTP as a request transmission protocol. »

Accordingly, the measures which are defined in ISO/IEC 25023 [160] cannot sufficiently measure accessibility in cloud environments. Therefore, regarding the mentioned definition of accessibility in cloud computing (that refers to all available cloud services and computing resources over network which can be accessed from anywhere [248, 249]), we propose "accessible cloud services and resources" as the third measure to evaluate this sub-characteristic in cloud environments more precisely. Since cloud environments are web-based and all the cloud resources through cloud services can be accessible remotely, thus, there is a possibility that certain resources or services due to some deficiencies in cloud services, may not be accessible. This inaccessibility, even for a short time, in some cases can cause the huge and irreparable damages, especially in the real-time cloud computing (such as intelligent transportation and traffic controlling). Based on this reason, the measure of «accessible cloud services and resources » is proposed. Subsequently, accessibility is an essential sub-characteristic of usability for three cloud service models and would be measurable for all types of cloud users.

Regarding the explanations in section 6.3.5 (continuity) and in the same vein in section 6.4.3 (environmental risk mitigation), to design the measurement method for the proposed measure of «accessible cloud services and resources», we adapt the precepts defined in ISO/IEC 25021

[232]. Thus, the information items to design a QME are provided as follows (adapted from ISO/IEC 25021 [232]) :

- QME name – Accessible cloud services and resources;
- Target entity – Cloud services and resources;
- Objectives and property to quantify – To know the degree of cloud services and resources accessibility in cloud environments;
- The properties that need to be measured are : 1- Number of required accessible cloud services and cloud resources in a specific time, 2- Number of required cloud services and cloud resources that need to be accessible in a specific time;
- Relevant quality measure(s) – Usability (characteristic level) and accessibility (sub-characteristic level) defined in ISO/IEC 25010 [1];
- Measurement method – The number of required accessible cloud services and cloud resources in a specific time is divided by the number of required cloud services and cloud resources that need to be accessible in a specific time to estimate the degree of accessibility of cloud services and cloud resources;
- Input for the QME – The required cloud services and resources;
- Unit of measurement for the QME - The number of accessible required cloud services and resources;
- Numerical rules – Dividing;
- Scale type – Ratio;
- Context of QME – This QME is usable to measure the accessibility of the cloud services;
- System life cycle process(es) – Operation and maintenance.

Table 6-24 shows the measures of accessibility in cloud environments.

Table 6-24. Accessibility measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Accessibility for users with disabilities	m_{21} = F/f F: Number of usable functions by the disable users f: Number of implemented functions	Doable	All	All
Supported languages adequacy	m_{22} = L/l L: Number of supported languages l: Number of languages needed to be supported	Doable	All	All
Accessible cloud services and resources*	m_{23}= A/a A: Number of required accessible cloud services and cloud resources in a specific time a: Number of required cloud services and cloud resources that need to be accessible in a specific time	Doable	All	All

6.5.7 Courtesy

In ISO/IEC 25011 [148], courtesy is defined as :

> «degree to which the IT service is provided in a polite, respectful and friendly way. »

Thus, this sub-characteristic refers to cloud components, interfaces and other elements that have interaction with the cloud service users. In this way, the represented massages from the system side to the users need to be politely provided. Otherwise, it may have negative impacts on adoption of the provided cloud services.

Since there is no defined measure for courtesy in the related ISO/IEC standards, we propose «polite and clear message adequacy » as its measure. This measure is selected as it is extracted from the definition of courtesy in ISO/IEC 25011 [148].

Regarding the explanations in section 6.3.5 (continuity) and in the same vein in section 6.4.3 (environmental risk mitigation) and section 6.5.6 (accessibility), to design the measurement method for the proposed measure of «polite message adequacy», we adapt the precepts defined in ISO/IEC 25021 [232]. Thus, the information items to design a QME are provided as follows (adapted from ISO/IEC 25021 [232]) :

- QME name – Polite message adequacy;
- Target entity – Cloud components, interfaces and other elements that have interaction with the cloud service users;
- Objectives and property to quantify – To know the degree of polite message adequacy in cloud services;
- The properties that need to be measured are : 1- The number of represented messages provided in a polite way, 2- The total number of represented messages;
- Relevant quality measure(s) – Usability (characteristic level) defined in ISO/IEC 25010 [1] and courtesy (sub-characteristic level) defined in ISO/IEC 25011 [148];
- Measurement method – The number of represented messages provided in a polite way is divided by the total number of represented messages;
- Input for the QME – The messages provided for the cloud services to be represented to the cloud users;
- Unit of measurement for the QME - The number of represented messages related to the cloud services;
- Numerical rules – Dividing;
- Scale type – Ratio;
- Context of QME – This QME is usable to measure the courtesy of the cloud services messages;
- System life cycle process(es) – Testing, implementation, operation and maintenance.

Accordingly, the proposed measure for courtesy (table 6-25), would be applicable for three cloud service models and can be measurable for all type of cloud users.

Table 6-25. Courtesy measure

Measures	Evaluation Function	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
Polite message adequacy	m_{24} = M/m M: The number of represented messages provided in a polite way m: The total number of represented messages	Doable	All	All

6.6 The verification of compatibility measures

As explained in chapter 4, section 4.1.8, in particular cases it is necessary to use various cloud services provided by different cloud service providers simultaneously in diverse platforms. Thus, compatibility can enable systems to provide adopters with various advantages such as reducing time and cost since cloud users can exclude costly infrastructure implementation [251, 252].

6.6.1 Co-Existence

To evaluate the ability of a cloud service to meet the requirements efficiently in a common shared environment with the shared resources simultaneously with other cloud services, the measures of co-existence are presented (ISO/IEC 25023 [160]). Taking into account the diversity of cloud users' requirements (for example, an organization needs to use PaaS and SaaS whether from the same cloud provider or different ones), evaluating this measure (table 6-26) would be essential for all cloud service models.

Table 6-26. Co-existence measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Co-existence with other cloud services	$m_1 = S/s$ S: Number of other specified cloud services with which this service can co-exist s: Number of other cloud services specified to co-exist with this service in the operation environment	Doable	All	Organization and Cloud Service Provider

6.6.2 Interoperability

Interoperability as a scientific challenge has been established since diversity of environments in which enterprises have collaborations with their customers and suppliers is increasing incrementally [253]. According to the definition of interoperability measures in ISO/IEC 25023 [160], interoperability in cloud computing alludes to the ability of sharing and exchanging information from one cloud provider to another or between private and public clouds and successfully using the exchanged information to meet the requirements [254-256]. Taking into account the key characteristics of cloud computing explained in ISO/IEC 17788 [24], such as on-demand self-service and resource pooling, interoperability needs to be considered inside the cloud, between clouds and between conventional and cloud-based systems [253]. The absence of interoperability can be originated through technological variety of cloud service platform offerings such as various virtualization technologies, various service interfaces and development environments [254, 257, 258]. Therefore, in the process of evaluating the trustworthiness in cloud environments to select a trustworthy cloud service provider, measuring interoperability in IaaS, PaaS and SaaS would be necessary.

Table 6-27. Interoperability measures

Measures (adapted from ISO/IEC 25023) [160]	**Evaluation Function (adapted from ISO/IEC 25023) [160]**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Data format exchangeability	$m_2 = D/d$ D: Number of data formats exchangeable with other cloud services d: Number of data formats specified to be exchangeable	Doable	All	Organization and Cloud Service Provider
Data exchange protocol sufficiency	$m_3 = E/e$ E: Number of supported data exchange protocols e: Number of data exchange protocols specified to be supported	Doable	All	Organization and Cloud Service Provider
External interface adequacy	$m_4 = I/i$ I: Number of functional external interfaces (interfaces with other software and systems) i: Number of specified external interfaces (interfaces with other software and systems)	Doable	All	All

6.7 The verification of IT service adaptability measures

As it is explained in ISO/IEC 25011 [148] and [162], adaptability in cloud computing is the cloud service capability to be adjusted depend on users' needs. Regarding the three cloud service models, hardware and software adaptability of a cloud service would be crucial to address diverse requirements.

6.7.1 Customizability

Customizability in ISO/IEC 25011 [148] is defined as :

> «degree to which the IT service can be customized at the request of users. »

Based on this definition, in a multi-tenant cloud environment with a wide variety of differences in users' requirements, the cloud resources need to be highly customizable [259]. Customizability in IaaS refers to the situations in which cloud user is able to deploy customized virtual appliances and has (root) access to the virtual servers [259, 260]. By contrast, PaaS and SaaS present less flexibility in customizability and are not convenient for general-purpose computing, but still are supposed to provide a certain customization level [259].

The reason of low-level customizability in SaaS and PaaS is that these services are easily accessible through Internet for the customers. Thus, since the number of potential users of these services is growing, the providers are not able to modify their cloud services [261, 262]. However, service customers are simply able to change or modify their services based on their requirements [261, 262]. If service providers do not offer customizable cloud services, that causes the limitation of the usages of the SaaS and PaaS services [261]. This fact, to some extent refers to the definition of operability in ISO/IEC 25010 [1] which is the sub-characteristic of usability (section 6.5) ;

> « Operability : degree to which a product or system has attributes that make it easy to operate and control. »

Accordingly, there is a close interconnection between operability and customizability as one of the sub-characteristics of IT service adaptability. Thus, the two presented measures of customizability in table 6-28, are adapted from operability (section 6.5.3, table 6-21) which are defined in ISO/IEC 25023 [160].

Table 6-28. Customizability measures

Measures (adapted from ISO/IEC 25023 [160])	**Evaluation Function (adapted from ISO/IEC 25023 [160])**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Functional customizability	$m_1 = F/f$ F: Number of customizable functions f: Number of functions for which users could benefit from customization	Doable	All	All

Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023 [160])	Type of category (Doable/ Undoable)	Applicable for (SaaS/PaaS/ IaaS)	Type of user (Individual/ Organization/ Cloud Service Provider)
User interface customizability	m_2 = H/h H: Number of customizable user interface elements h: Number of user interface elements for which users could benefit from customization	Doable	All	All

6.7.2 Initiative

Initiative in ISO/IEC 25011 [148] is defined as the degree to which the service is able to discover users' goal and propose modifications to meet users' requirements. However, initiative is not fully discussed in the literature as a sub-characteristic of IT service adaptability in cloud computing. Regarding initiative definition, it is recommended to be evaluated for all cloud service models. Since it seems it can be helpful for users to fulfill their goals in the use of cloud services.

Since there is no defined measure for initiative in related ISO/IEC standards, we propose «initiative adequacy» as its measure. This measure is selected as it is extracted from the definition of initiative in ISO/IEC 25011 [148].

Regarding the explanations in section 6.3.5 (continuity) and in the same vein in section 6.4.3 (environmental risk mitigation) and section 6.5.6 (accessibility) and section 6.5.7 (courtesy), to design the measurement method for the proposed measure of «initiative adequacy», we adapt the precepts defined in ISO/IEC 25021 [232]. Thus, the information items to design a QME are provided as follows (adapted from ISO/IEC 25021 [232]) :

- QME name – Initiative adequacy;
- Target entity – Cloud components, interfaces and other elements that have interaction with the cloud service users;

- Objectives and property to quantify – To know the degree of initiative adequacy in cloud services;
- The properties that need to be measured are : 1- Number of services that have initiative ability, 2- Number of services that could benefit from initiative ability;
- Relevant quality measure(s) – IT service adaptability (characteristic level) and initiative (sub-characteristic level) defined in ISO/IEC 25011 [148];
- Measurement method – The number of services that have initiative ability is divided by the total number of services that could benefit from initiative ability;
- Input for the QME – Cloud components, interfaces and other elements that have interaction with the cloud service users;
- Unit of measurement for the QME - The number of cloud components, interfaces and other elements that have interaction with the cloud service users;
- Numerical rules – Dividing;
- Scale type – Ratio;
- Context of QME – This QME is usable to measure the ability of initiative in cloud services;
- System life cycle process(es) – Testing, implementation, operation and maintenance.

Table 6-29 presents the proposed measure for initiative which can be measurable for all types of cloud user.

Table 6-29. Initiative measures

Measure	**Evaluation Function**	**Type of category (Doable/ Undoable)**	**Applicable for (SaaS/PaaS/ IaaS)**	**Type of user (Individual/ Organization/ Cloud Service Provider)**
Initiative adequacy	$m_1 = D/d$ D: Number of services that have initiative ability d: Number of services that could benefit from initiative ability	Doable	All	All

6.8 Conclusion

This chapter set out with the aim of verifying the applicability of the measures of trust characteristics in cloud environments. There are two main reasons for this verification:

- As these measures are mainly extracted from system and software quality standards (ISO/IEC 25023 [160], ISO/IEC 25022 [173] and to some extent based on the definitions in ISO/IEC 25011[148]), thus, it is necessary to ensure the applicability of these measures in cloud computing;
- According to the previous explanations, since the users' level of access to cloud resources in terms of each cloud service model is different, thus, it is essential to clarify the feasibility of evaluation for each type of cloud service users.

In addition, in this chapter we consider three principles:

- Data: which is the precious asset of different cloud service model users (individual, organization and provider) needs to be preserved from unwanted modifications, retrieved correctly and exchanged securely in a specified time. We believe that each of the cloud service models has some valuable data, needs to work with especial data and produces different data;
- Subjective and Objective trust: in this chapter, the measures for subjective trust (i.e. related to the users' experience with the cloud services through customer support policy) and the measures for objective trust (i.e. related to the quality of services such as performance efficiency) are considered. According to the tables discussed in this chapter, we proposed various measures for these two types of trust in cloud environments in order to evaluate trust;
- Addressing requirements: a cloud user with the hope of surmounting the needs, adopts one or more types of cloud service models (IaaS, PaaS, SaaS). Thus in this study we tried to propose different characteristics in order to create a reference from users' requirements as complete as possible to be measured in cloud computing.

In addition, we develop some measures for the characteristics for which there is no defined measures in the related ISO/IEC standards, based on the precepts discussed in ISO/IEC 25021 [232].

The results of this chapter and the previous chapter which are the stages of tracing phase are the separate models of measurable trust characteristics for the three discussed categories of cloud service users that are explained in the next chapter. These results will be applied in the trust evaluation process by the proposed cloud trust model (ED-BeCT).

CHAPTER 7

THE RESULTS OF TRACING PHASE

In this chapter, we elaborate on the results of tracing phase which encompasses two stages : 1- an investigation into IaaS, PaaS and SaaS, 2- Verifying the applicability of the measures. These two stages are discussed in chapters 5 and 6 respectively.

As a summary of these stages, we have investigated into cloud service models to discover their main features, their similarity and difference to be able to verify the applicability of identified measures in cloud computing. Considering this explanation, the reason that why we entitle this phase of the methodology to « tracing phase » can be justified.

7.1 The separate models of measurable trust characteristics for users

The obtained results from analyzing the literature (in chapter 1), and chapters 4, 5 and 6 which are associated with trust characteristics, their measures and verifying the applicability of their measures in cloud environments respectively, are adequate evidence to show that trust in cloud computing needs to include several characteristics. These characteristics along with all their sub-characteristics and measures are the representatives of a great variety of users' requirements, quality of services, different aspects of objective and subjective trust related to the cloud services and also different key intrinsic characteristics of cloud computing which are defined in ISO/IEC 17788 [24]. Although, most of these characteristics are quality-based and are derived from software quality standards, to distinguish them from conventional software quality characteristics (given the cloud computing definition in NIST [2]), and also to emphasize on the fact that trust is not just quality of the cloud services, they are named as «trust characteristics » in cloud computing.

These trust characteristics along with their measures discussed in chapter 6 and preceding chapters are the important factors of a trustworthy cloud service provider and by assessing cloud services from various aspects, can provide a flexible mechanism in addressing different

requirements. The results of trust characteristics evaluation can create a positive declaration intended to give confidence and augment users' assurance to have a healthy collaboration with cloud service providers.

Moreover, chapter 6 indicates that the specified measures of trust characteristics can be categorized into two groups : doable and undoable from the perspective of feasibility of measuring in cloud environments. There exist two basic reasons for that :

- As already discussed in chapter 6, the major similarity among IaaS, PaaS and SaaS can be provisioning of the cloud resources in a flexible and abstracted way [199]. Given that the identified trust characteristics and their measures are generalized for three cloud service models (IaaS, PaaS, SaaS) and based on the fact that cloud users have various precious data to migrate in cloud environments with the aim of applying cloud resources according to their needs, therefore, the reason that why most of the proposed measures are applicable to be evaluated in cloud environments can be clearly justified;
- On the other hand, among the proposed measures there are certain measures that all the three presented categories of cloud users cannot measure them based the identified measures extracted from relevant standards. As an example, we can refer to capacity measures presented in the table 6-3 in chapter 6 as it has three measurable measures. In this case, although all the three categories of users are able to measure its first measure (transaction/job processing capacity), its two other measures are measurable only for cloud service providers. The main reason regarding the details of the cloud service models discussed in chapter 5 and also explained in [153], lies in the key difference among IaaS, PaaS and SaaS which is the level of access to cloud resources (figure 5-3 in chapter 5). Therefore, based on the explained categories of cloud users (individual, organization, cloud service provider that is the owner of the cloud services), the level of access to cloud resources can influence on the measuring processes based on the measurement functions presented in the tables 1 to 29 in previous chapter, in particular the measures categorized into the « doable » group.

Although, by various performed analyses in preceding chapters we generalized these trust characteristics and their measurable measures to be applicable in three cloud service models (IaaS, PaaS, SaaS), based on the level of access to the cloud resources, it is necessary to particularize the proposed set of trust characteristics in cloud computing (figure 3-4 in chapter 3) for the three specified categories of cloud service users. Thereby, individuals, organizations and also the owners of the cloud services (providers), can have practical measurements according to their related models. These models are explained in the following sections.

7.1.1 Individuals

Figure 7-1 illustrates the identified model for the category of individual from cloud service users. The presented model in figure 7-1 is categorized into 7 characteristics, with some of them decomposed into sub-characteristics. As explained in chapter 4 and based on the ISO/IEC 17788 [24], multi-tenancy is one of the key characteristics of cloud computing and auditability is one of the key cross-cutting aspects of cloud computing and based on the identified trust tree in chapter 4 (figure 4-1) and its related explanations, the importance of these two characteristics can be verified. Compared to the general model in chapter 3 (figure 3-4) and referring to the discussed tables 1 to 29 in chapter 6, it can be noticed that compatibility, freedom from risk, rapid scalability and elasticity are omitted. Since the measures of their sub-characteristics (partially or completely) may not be measurable for individuals in cloud environments. The reason of partially or completely not measurable is that for example, compatibility has two sub-characteristics and each of them includes one or more measures. As table 6-26 in previous chapter presents, there is no measurable measure for the first sub-characteristic of compatibility (completely) but for the second sub-characteristic (table 6-27 in chapter 6) there is only one measurable measure for individuals (partially). This creates a gap in the measurement process, as it is recommended to consider the sub-characteristics along with all their measures to evaluate the root characteristic. In the same vein, there are some removed sub-characteristics, as all their measures are not measurable for individual category of cloud users. Thus, excluding them might provide more credible and comprehensive results.

To summarize, the main criteria for adding characteristics and sub-characteristics in the model are explained respectively, as follows :

- The related sub-characteristics of a root characteristic must be measurable with all the identified measures;
- All the measures of a sub-characteristic must be measurable for the individual category.

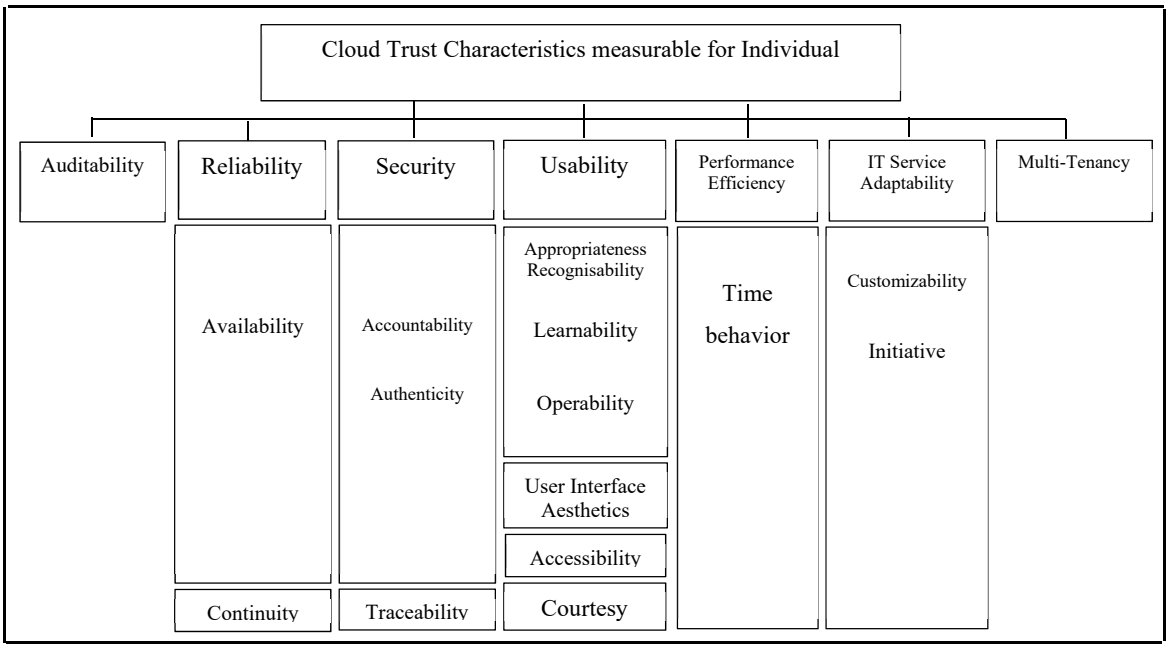

Figure 7-1. Cloud Trust Characteristics measurable for Individual

7.1.2 Organization

Figure 7-2 depicts the identified model for organization category of cloud service users. This model is decomposed into 9 characteristics along with their sub-characteristics. Followed by the determined rules explained for figure 7-1, in this model also rapid scalability and elasticity are omitted. In addition, compaired to the figure 3-4 (in chapter 3) some of the sub-characteristics are excluded from this model since their measures are not measurable for this category of cloud users. Based on figure 6-1 and its explanations in chapter 6, as the category

of organization possesses a wider level of access to the cloud resources, thus, there are more measurable characteristics and sub-characteristics added in its model (figure 7-2) in comparison to the individual model (figure 7-1).

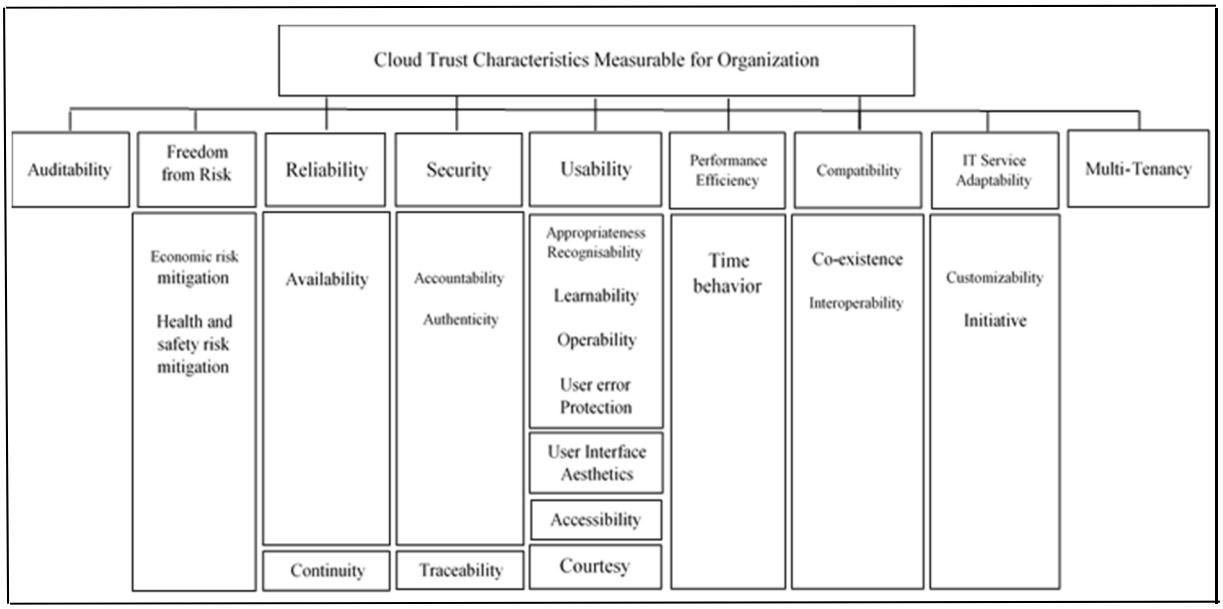

Figure 7-2. Cloud Trust Characteristics Measurable for Organization

7.1.3 Cloud service provider (the owner of the cloud services)

Figure 7-3 presents the identified model for the category of cloud service provider of the cloud users. Considering the previous explanations related to figure 6-1 in chapter 6, it is almost certain that cloud service providers have the broadest level of access to cloud resources in comparison to individuals and organizations. Accordingly, regarding tables 1 to 29 in previous chapter, this category of cloud users is believed to be able to measure all the measurable characteristics and sub-characteristics, as they possess cloud resources and their configurations methods, policy of providing cloud services and related confidential information which are kept secret from other categories of cloud users. Obviously, the presented model in figure 7-3 contains all the measurable characteristics along with their sub-characteristics. The results of evaluating cloud services based on these characteristics and sub-characteristics according to

their identified measures can be helpful for this category of cloud users for the purposes of modifications and improvements of the provided cloud services.

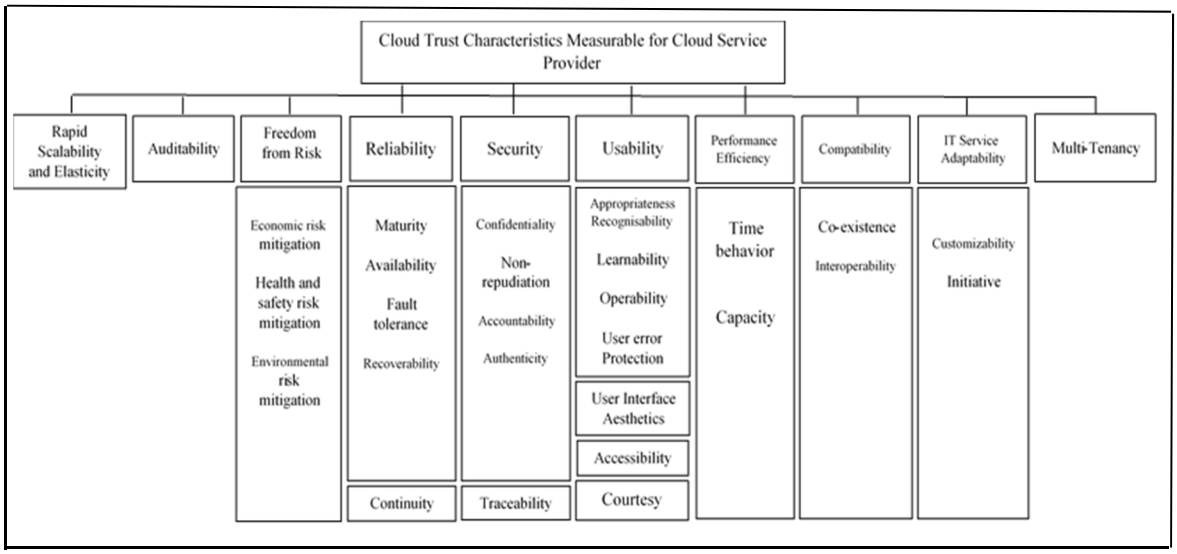

Figure 7-3. Cloud Trust Characteristics Measurable for Cloud Service Provider

The discussed models (figures 7-1, 7-2 and 7-3) stem from the figure 3-4 in chapter 3 and are as the results of verifying the applicability of the identified measures in cloud environments in previous chapter. Accordingly, the goals of presenting these models are explained as follows :

- These models provide a more accurate insight into the importance of cloud users' roles in cloud environments and their level of access to cloud resources in assessing cloud services and more precisely evaluating trust in cloud computing;
- Each category of cloud users (individual, organization, cloud service provider) possesses a separate model of measurable trust characteristics and sub-characteristics to facilitate identifications of these characteristics to be evaluated in the processes of trust assessment in cloud computing based on cloud users' requirements with ED-BeCT (chapter 8).

7.2 The measures of trust characteristics presented for each category

In this section, based on the presented models for individual, organization and cloud service provider we present the equations of the trust characteristics in the following tables (1 to 7). In these tables, we omit the sub-characteristics for which even one of its measures, is not measurable for each category of cloud user. In all the tables in this chapter, Ind, Org and CSP stand for individual, organization and cloud service provider respectively.

7.2.1 Performance efficiency measures

The measurable measures for the sub-characteristics of performance efficiency along with the evaluation functions are illustrated in the table 7-1.

Table 7-1. Performance efficiency evaluation criteria

Sub-Characteristics	The measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Time Behavior	Mean response time	$m_1 = \sum_{i=1}^{n} \frac{T_i}{n}$ (1) T_i: Response time to a user task or system task at i-th evaluation. n: Total number of response evaluations	✓	✓	✓
	Response time adequacy	$m_2 = T/t$ T: Mean response time evaluated from (1) t: The specified target response time	✓	✓	✓
	Mean turnaround time	$m_3 = \sum_{i=1}^{n}(T_i - C_i)/n$ (2) T_i: Starting time of a job i C_i: The completion time of the job i n ; Number of evaluations	✓	✓	✓
	Turnaround time adequacy	$m_4 = T/t$ T: Mean turnaround time evaluated by (2) t: The specified target turnaround time	✓	✓	✓
	Mean throughput	$m_5 = \sum_{i=1}^{n}(\frac{J_i}{O_i})/n$ J_i: The number of completed jobs during the i – th observation time O_i: i-th observation time period n: The number of observations	✓	✓	✓

Sub-Characteristics	The measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Capacity	Transaction processing capacity	$m_6 = T/t$ T: The number of completed transactions in observation time t: Observation time	✓	✓	✓
	User access capacity	$m_7 = \sum_{i=1}^{n} \frac{T_i}{n}$ T_i: The maximum number of users who can simultaneously access the system at i-th observation n: The number of observations			✓
	User access increase adequacy	$m_8 = U/t$ U: The number of users added in observation time t: Observation time			✓
Elasticity (ISO/IEC 19086 -1 [150])	Speed*	$m_9 = (R+C)/n$ R: The total number of time lasted to resource re-allocation C: The total number of time lasted to change work load n: The total number of requests			✓
	Provision*	$m_{10} = R - N$ R: The number of actually allocated resources N: Total number of needed resources			✓

In table 7-1, the sub-characteristics of performance efficiency with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.2 Security measures

The measurable measures for the sub-characteristics of security along with the evaluation functions are illustrated in the table 7-2.

Table 7-2. Security evaluation criteria

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Confidentiality	Access controllability	$m_1 = 1 - D/d$ D: The number of confidential data items that can be accessed without authorization d: The number of data items that require access control	✓	✓	✓
	Data encryption correctness	$m_2 = D/d$ D: The number of data items encrypted or decrypted correctly d: The number of data items that require encryption or decryption			✓
	Strength of cryptographic algorithms	$m_3 = 1 - A/a$ A: The number of cryptographic algorithm broken or unacceptably risky in use a: The number of used cryptographic algorithms			✓
Non-Repudiation	Digital signature usage	$m_4 = E/e$ E: The number of events that ensure non-repudiation using digital signature e: The number of events which require non-repudiation by using digital signature			✓
Accountability	User audit trail completeness	$m_5 = A/a$ A: The number of accesses recorded in all logs a: The number of accesses to tested system or data	✓	✓	✓
	System log retention	$m_6 = T/t$ T: Duration for which the system log is retained in stable storage t: Specified retention period to keep the system log in stable storage	✓	✓	✓
Authenticity	Authentication mechanism sufficiency	$m_7 = A/a$ A: The number of provided authentication mechanisms a: The number of specified authentication mechanisms	✓	✓	✓
	Authentication rules conformity	$m_8 = R/r$ R: The number of implemented authentication rules	✓	✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		r: The number of specified authentication rules			
Traceability (ISO/IEC 25011 [148])	Traceable outcomes*	m_9 = O/o O: The number of traceable outcomes to/from the user needs o: The total number of outcomes to/from the user needs	✓	✓	✓

In table 7-2, the sub-characteristics of security with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.3 Measuring multi-tenancy and rapid scalability and elasticity

According to the explanations and regarding the trust tree in figure 4-1 in chapter 4, it can be found that multi-tenancy can be measured by evaluating performance efficiency, security, scalability and elasticity. In addition, to evaluate scalability and elasticity, performance efficiency with all its measures is proposed to be its reference sub-characteristic.

7.2.4 Reliability measures

The measurable measures for the sub-characteristics of reliability along with the evaluation functions are illustrated in the table 7-3.

Table 7-3. Reliability evaluation criteria

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Maturity	Fault correction	$m_1 = F/f$ F: The number of corrected reliability-related faults f: The number of detected reliability-related faults			✓
	Mean time between failure	$m_2 = O/f$ O: Operation time f: The number of occurred failures	✓	✓	✓
	Failure rate	$m_3 = F/t$ F: The number of detected failures in observation time t: Observation time	✓	✓	✓
	Test coverage	$m_4 = F/f$ F: Number of performed system or software functions f: Number of system or software functions included in their associated test suits			✓
Availability	System availability	$m_5 = T/t$ T: Provided system operation time t: Specified system operation time	✓	✓	✓
	Mean down time	$m_6 = T/b$ T: Total down time b: The number of observed breakdowns	✓	✓	✓
Fault Tolerance	Failure avoidance	$m_7 = F/f$ F: The number of avoided critical failure (based on test cases) f: The number of executed test cases of fault pattern during testing	✓	✓	✓
	Redundancy of components	$m_8 = C/c$ C: Number of system components redundantly installed c: Number of system components			✓
	Mean fault notification time	$m_9 = \sum_{i=1}^{n}(T_i - C_i)/n$ T_i: The time at which the fault i is reported by the system C_i: The time at which fault i is detected			✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		n: Number of faults detected			
Recoverability	Mean recovery time	$m_{10} = \sum_{i=1}^{n} T_i / n$ T_i: Total time to recover the downed software/system for each failure i n: Number of failures	✓	✓	✓
	Backup data completeness	m_{11} = D/d D: The number of backed up data items d: The total number of data which require backup			✓
Continuity (ISO/IEC 25011 [148])	Customer support policy*	m_{12} = S/s + B/b + F/f S: The number of supported foreseeable circumstances to mitigate the risks resulting from interruption to an acceptable level s: The total number of foreseeable circumstances B: Number of fixed bugs/errors in a specific time b: The total number of informed bugs in a specific time F: The number of added new features in a specific time f: Total number of required features in a specific time	✓	✓	✓

In table 7-3, the sub-characteristics of reliability with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.5 Freedom from risk measuring

The measurable measures for the sub-characteristics of freedom from risk along with the evaluation functions are illustrated in the table 7-4.

Table 7-4. Freedom from risk evaluation criteria

Sub-Characteristics	The Measures (adapted from ISO/IEC 25022 [173])	Evaluation Function (adapted from ISO/IEC 25022 [173])	Doable for		
			Ind	Org	CSP
Economic Risk Mitigation	Return on investment (ROI)	$m_1 = (B - I)/I$ B: Additional benefits obtained I: Invested amount		✓	✓
	Time to achieve return on investment	$m_2 = T$ T: Time to achieve ROI		✓	✓
	Business performance	$m_3 = Aa/At$ A: Profitability or sales of the company (a: actual, t: target)		✓	✓
	Benefits of IT investment	$m_4 = Ba/Bt$ B: The benefit of IT investment (a: actual, t: target)		✓	✓
	Service to customers	$m_5 = S/s$ S: Actual level of service s: Intended level of service	✓	✓	✓
	Cloud customers loyal to a specific cloud service provider	$m_6 = L/C$ L: The number of the loyal customers in a specified time C: The total number of the customers in a specified time		✓	✓
	Revenue from each customer	$m_7 = R$ R: Revenue from a customer		✓	✓
	Errors with economic consequences	$m_8 = E/U$ E: The number of errors with economic consequences U: The total number of usage situations		✓	✓
Health and Safety Risk Mitigation	User health reporting frequency	$m_9 = U/u$ U: Number of users reporting health problems u: Total number of users		✓	✓
	User health and safety impact	$m_{10} = \frac{1}{T_b}\sum_{i=1}^{n}(T_{a_i} \times S_i)$ n: Number of affected people Ta_i: Length of time for which the i-th person is effected S_i: Degree of significance of the impact on the i-th person T_b: Length of time from start of system in operation		✓	✓
	Safety of people affected by use of the system	$m_{11} = P/T$ P: The number of people put at hazard		✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25022 [173])	Evaluation Function (adapted from ISO/IEC 25022 [173])	Doable for		
			Ind	Org	CSP
		T: Total number of people who could be affected by use of the system			
Environmental risk mitigation	Power* consumption	m_{12} = Aa/At A = Power consumption (a: actual, t:target)			✓

In table 7-4, the sub-characteristics of freedom from risk with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.6 Usability measures

The measurable measures for the sub-characteristics of usability along with the evaluation functions are illustrated in the table 7-5.

Table 7-5. Usability evaluation criteria

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Appropriateness Recognisability	Description completeness	m_1 = S/s S: The number of usage scenarios described in the product documents s: The number of usage scenarios of the product	✓	✓	✓
	Demonstration coverage	m_2 = T/t T: The number of tasks with demonstration features t: The number of tasks that could benefit from demonstration features	✓	✓	✓
	Entry point self-descriptiveness	m_3 = S/s S: The number of landing pages of user console that explain the purpose of service	✓	✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		s: The number of landing pages of user console in a service			
Learnability	User guidance completeness	$m_4 = F/f$ F: The number of functions described in user documentation f: The number of implemented functions	✓	✓	✓
	Entry fields defaults	$m_5 = E/e$ E: The number of entry fields whose default values have been automatically filled in during operation e: The number of entry fields that could have default values	✓	✓	✓
	Error message understandability	$m_6 = M/m$ M: Number of error messages which state the reason of occurrence and suggest the ways of resolution m: Number of implemented error messages	✓	✓	✓
	Self-explanatory user interface	$m_7 = I/i$ I: Number of information elements and steps that are presented in a way that the user could understand i: Number of information elements and steps needed to complete common tasks for a first time user	✓	✓	✓
Operability	Operational consistency	$m_8 = 1 – T/t$ T: Number of specific interactive tasks that are performed inconsistently t: Number of specific interactive tasks that need to be consistent	✓	✓	✓
	Message clarity	$m_9 = M/m$ M: Number of messages that convey the right outcome or instructions to the user m: Number of messages implemented	✓	✓	✓
	Functional customizability	$m_{10} = F/f$ F: Number of customizable functions	✓	✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		f: Number of functions which users could benefit from customization			
	User interface customizability	m_{11} = I/i I: Number of customizable user interface elements i: Number of user interface elements that could benefit from customization	✓	✓	✓
	Monitoring capability	m_{12} = F/f F: Number of functions having state monitoring capability f: Number of functions that could benefit from monitoring capability	✓	✓	✓
	Undo capability or re-confirmation	m_{13} = T/t T: Number of tasks that provide undo capability or re-confirmation t: Number of tasks for which users could benefit from having re-confirmation or undo capability	✓	✓	✓
	Understandable categorization of information	m_{14} = I/i I: Number of information structures familiar for users i: Number of used information structures	✓	✓	✓
	Appearance consistency	m_{15} = 1 – U/u U: Number of user interfaces with similar items but with different appearances u: Number of user interfaces with similar items	✓	✓	✓
	Input device support	m_{16} = T/t T: Number of tasks that can be initiated by all appropriate input modalities t: Number of tasks supported by the system	✓	✓	✓
User Error Protection	Avoidance of user operation errors	m_{17} = A/a A: Number of user actions and inputs that are protected from causing any system malfunction	✓	✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		a: Number of user actions and inputs that could be protected from causing any system malfunction			
	User entry error correction	m_{18} = E/e E: Number of entry errors for which the system provides a suggested correct value e: Number of entry errors detected	✓	✓	✓
	User error recoverability	m_{19} = E/e E: Number of user errors that are designed and tested to be recovered by the system e: Number of user errors which can occur during operation	✓	✓	✓
User Interface Aesthetics	Appearance aesthetics of user interfaces	m_{20} = D/d D: Number of display interfaces aesthetically pleasing to the users in appearance d: Number of display interfaces	✓	✓	✓
Accessibility	Accessibility for users with disabilities	m_{21} = F/f F: Number of usable functions by the disable users f: Number of implemented functions	✓	✓	✓
	Supported languages adequacy	m_{22} = L/l L: Number of supported languages l: Number of languages needed to be supported	✓	✓	✓
	accessible cloud services and resources*	m_{23}= A/a A: Number of required accessible cloud services and cloud resources in a specific time a: Number of required cloud services and cloud resources that need to be accessible in a specific time	✓	✓	✓
Courtesy (ISO/IEC 25011 [148])	Polite messages adequacy*	m_{24} = M/m M: The number of represented messages provided in a polite way	✓	✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
		m: The total number of represented messages			

In table 7-5, the sub-characteristics of usability with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.7 Compatibility measures

The measurable measures for the sub-characteristics of compatibility along with the evaluation functions are illustrated in the table 7-6.

Table 7-6. Compatibility evaluation criteria

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
Co-Existence	Co-existence with other cloud services	$m_1 = S/s$ S: Number of other specified cloud services with which this service can co-exist s: Number of other cloud services specified to co-exist with this service in the operation environment		✓	✓
Interoperability	Data format exchangeability	$m_2 = D/d$ D: Number of data formats exchangeable with other cloud services d: Number of data formats specified to be exchangeable		✓	✓
	Data exchange protocol sufficiency	$m_3 = D/d$ D: Number of supported data exchange protocols d: Number of data exchange protocols specified to be supported		✓	✓

Sub-Characteristics	The Measures (adapted from ISO/IEC 25023 [160])	Evaluation Function (adapted from ISO/IEC 25023) [160]	Doable for		
			Ind	Org	CSP
	External interface adequacy	m_4 = I/i I: Number of functional external interfaces I: Number of specified external interfaces	✓	✓	✓

In table 7-6, the sub-characteristics of compatibility with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.8 IT service adaptability measures

The measurable measures for the sub-characteristics of IT service adaptability along with the evaluation functions are illustrated in the table 7-7.

Table 7-7. IT service adaptability evaluation criteria

Sub-Characteristics	The Measures (ISO/IEC 25023 [160])	Evaluation Functions (ISO/IEC 25023 [160])	Doable for		
			Ind	Org	CSP
Customizability	Functional customizability	m_1 = F/f F: Number of customizable functions f: Number of functions for which users could benefit from customization	✓	✓	✓
	User interface customizability	m_2 = H/h H: Number of customizable user interface elements h: Number of user interface elements for which users could benefit from customization	✓	✓	✓
Initiative	Initiative adequacy*	m_3 = D/d D: Number of services that have initiative ability	✓	✓	✓

Sub-Characteristics	The Measures (ISO/IEC 25023 [160])	Evaluation Functions (ISO/IEC 25023 [160])	Doable for		
			Ind	Org	CSP
		d: Number of services that could benefit from initiative ability			

In table 7-7, the sub-characteristics of IT service adaptability with their measures are presented. In addition, the feasible assessment of the presented measures based on the explanations in chapter 6 is depicted for each category of cloud user in this table.

7.2.9 Auditability measures

Regarding the presented trust tree and its related explanations (figure 4-1 in chapter 4), we can consider that auditability will be measured by measuring the previous characteristics.

7.3 Chapter summary

This chapter summarized the results of the two stages of tracing phase in chapters 5 and 6 and has shown that the level of access to the cloud resources in the trust evaluation process is an important principle. However, this principle to some extent is ignored as there is no explanation in the literature.

Based on the presented features of IaaS, PaaS and SaaS in chapter 5 and the introduced categorization of cloud users in chapter 6, in this chapter we suggested three separate models of trust characteristics for the three groups of users : individual, organization and cloud service provider that is the owner of the cloud services. The discussed models (figures 7-1, 7-2 and 7-3) stem from figure 3-4 in chapter 3 and are as the results of verifying the applicability of the identified measures in cloud environments in previous chapter. Accordingly, the goals of presenting these models are explained as follows :

- These models provide a more accurate insight into the importance of cloud users' role in cloud environments and their level of access to cloud resources for assessing cloud services and more precisely evaluating trust in cloud computing;
- Each category of cloud users (individual, organization, cloud service provider) possesses a separate model of measurable trust characteristics and sub-characteristics to facilitate identifications of these characteristics to be evaluated in the process of trust assessment in cloud computing based on cloud users' requirements by ED-BeCT (chapter 8).

By this proposition, we obtain the following achievements:

- the trust evaluation process in cloud environments is feasible and easy;
- the proposed model in the current research project (ED-BeCT) is practical;
- the complexity of the model for each category of cloud users is decreased.

The next chapter is dedicated to the details of ED-BeCT and the process of applying these results.

CHAPTER 8

ED-BeCT : ENHANCED DYNAMIC-BEHAVIORAL CLOUD TRUST MODEL

As cloud computing is an evolving technology and regarding the fact that the number of cloud service suppliers is more and more increasing, the competition to gain market share is significant and on the upswing. Thus, the providers need to improve their services and apply up-to-date technologies to be able to stay on the market summit. Consequently, selection of a trustworthy cloud service provider regarding their close competitions may create some challenges. But the phases of ED-BeCT are tailored to provide a strong bridge to traverse these issues.

As discussed in chapter 1, there is a large volume of published studies describing various methods and proposing several models to evaluate trust in cloud computing based on the certain trust characteristics. Whilst this research project provide an enhanced model by considering a set of trust characteristics and sub-characteristics along with their measures that is able to dynamically measure the actual behavior of the cloud services and compare it with the proposed service level agreement (SLA). The proposed model divides the trust characteristics related to the each category of cloud users into two groups : characteristics for filtering and characteristics for selecting.

The details of this methodology and description of the phases of ED-BeCT to evaluate the provided cloud services are presented in this chapter. Based on the results obtained by applying this model, the trustworthy cloud service provider can be selected.

8.1 The methodology of ED-BeCT

The methodology of ED-BeCT consists of nine phases which are explained as follows:

Phase 1. Identification of trust characteristics for CSP filtering

As discussed in chapter 6, to clarify the feasibility of the trust characteristics evaluation based on the identified measures, it is necessary to specify the cloud users' access level to the cloud resources. Therefore, cloud users are categorized into three categories :

- Individual : is the end user of the cloud services;
- Organizations : cloud service providers that have collaboration together;
- Cloud service provider (CSP) : the owner of the cloud services.

Accordingly in this phase, first it is essential to specify the business goals for using cloud services. Based on the specified business goals, cloud users given that in which of the aforementioned categories may be accommodated, should negotiate with the cloud service providers to identify the characteristics in the proposed service level agreement (SLA). The models that are discussed in the previous chapter in the figures 7-3, 7-4 and 7-5 for the three categories of cloud users (i.e. individual, organization and CSP respectively) can be helpful in this regard to have an insight for the measurable trust characteristics by each category of cloud users. Although, the proposed SLA for each cloud service provider may be different, there are certain characteristics such as the characteristics related to the quality of services (QoS) which are common in all SLAs. The values of these characteristics in the related SLAs are considered as ideal values for the trust characteristics. In certain cases that there is no common characteristics among the offered SLAs, the values mentioned in each SLA for the characteristics will be considered.

In this phase, cloud users regarding the reputation of the cloud providers and by taking into account the feedback and recommendations of the other users, can limit the number of providers to negotiate. As reputation can be achieved by various ways and feedback and recommendations are generally based on personal idea (because each cloud user has particular requirements and based on these requirements, their feedback may differ), thus we do not

consider them in our measurements. However, they can influence on the beginning of the evaluation process in the first phase in CSP identification.

The result of this phase would be a list of ideal measurable trust characteristics or sub-characteristics related to the required cloud services with their values that are specified by CSPs, mentioned in the proposed SLA of each cloud service provider.

It should be noted that throughout this chapter, the term « cloud service user(s) » or « user(s) », will be used in their broadest sense to refer to their discussed three categories (i.e. individual, organization and CSP).

Phase 2. Measuring (common) trust characteristics mentioned in SLAs

So far, users have identified their reference business goals and based on these business goals the (common) measurable trust characteristics and sub-characteristics in negotiated SLAs from the three models of individual, organization and cloud service provider (chapter 7) are specified. In this phase, users by applying the evaluation functions which are mainly extracted from system and software quality standards and discussed in chapter 6, would be able to measure these characteristics for the cloud services of the cloud providers with which users had negotiations in phase 1.

It should be noted that the evaluation functions discussed in the previous chapter are not presented in the current chapter to avoid repetition. In addition, based on the presented models for individual, organization and cloud service provider and the discussions in chapter 7 in the equations of the trust characteristics, we do not consider the sub-characteristics for which even one of its measures, is not measurable for each category of cloud user. In all the following equations, j indicates the measure presented in the related table of each trust characteristic in the previous chapter.

Measuring performance efficiency for the offered cloud services

Based on table 7-1 in chapter 7, performance efficiency of a cloud service provider can be measured by equations 8.1-8.3 which are related to the categories of individual, organization and cloud service provider respectively.

Performance efficiency with measurable measures for individuals :

$$E_{P_i} = \sum_{j=1}^{5} m_j \quad (8.1)$$

Performance efficiency with measurable measures for organization :

$$E_{P_o} = \sum_{j=1}^{5} m_j \quad (8.2)$$

Performance efficiency with measurable measures for cloud service provider :

$$E_{P_c} = \sum_{j=1}^{10} m_j \quad (8.3)$$

Measuring security for offered cloud services

Based on table 7-2 in chapter 7, security of a cloud service provider can be measured by equations 8.4-8.6 which are related to the categories of individual, organization and cloud service provider respectively.

Security with measurable measures for individuals :

$$E_{S_i} = \sum_{j=5}^{9} m_j \quad (8.4)$$

Security with measurable measures for organization :

$$E_{S_o} = \sum_{j=5}^{9} m_j \quad (8.5)$$

Security with measurable measures for cloud service provider :

$$E_{S_c} = \sum_{j=1}^{9} m_j \quad (8.6)$$

Measuring multi-tenancy and rapid scalability and elasticity for offered cloud services

According to the explanations and regarding the trust tree in figure 4-1 in chapter 4, it can be found that multi-tenancy can be measured by evaluating performance efficiency, security, scalability and elasticity. In addition, to evaluate scalability and elasticity, performance efficiency with all its measures is a reference sub-characteristic.

Measuring reliability for offered cloud services

Based on table 7-3 in chapter 7, reliability of a cloud service provider can be measured by equations 8.7-8.9 which are related to the categories of individual, organization and cloud service provider respectively.

Reliability with measurable measures for individuals :

$$E_{R_i} = \sum_{j=5}^{6} m_j + m_{12} \quad (8.7)$$

Reliability with measurable measures for organization :

$$E_{R_o} = \sum_{j=5}^{6} m_j + m_{12} \quad (8.8)$$

Reliability with measurable measures for cloud service provider :

$$E_{R_c} = \sum_{j=1}^{12} m_j \quad (8.9)$$

Measuring freedom from risk for offered cloud services

Based on table 7-4 in chapter 7, freedom from risk of a cloud service provider can be measured by equations 8.10-8.11 which are related to the categories of organization and cloud service provider respectively.

Freedom from risk with measurable measures for organization :

$$E_{F_o} = \sum_{j=1}^{11} m_j \quad (8.10)$$

Freedom from risk with measurable measures for cloud service provider :

$$E_{F_c} = \sum_{j=1}^{12} m_j \quad (8.11)$$

Measuring usability for offered cloud services

Based on table 7-5 in chapter 7, usability of a cloud service provider can be measured by equations 8.12 which is common between the categories of individual, organization and cloud service provider :

$$E_U = \sum_{j=1}^{24} m_j \quad (8.12)$$

Measuring compatibility for offered cloud services

Based on table 7-6 in chapter 7, compatibility of a cloud service provider can be measured by equations 8.13 which is common between the categories of individual, organization and cloud service provider :

$$E_C = \sum_{j=1}^{4} m_j \tag{8.13}$$

Measuring IT service adaptability for offered cloud services

Based on table 7-7 in chapter 7, IT service adaptability of a cloud service provider can be measured by equations 8.14 which is common between the categories of individual, organization and cloud service provider :

$$E_I = \sum_{j=1}^{3} m_j \tag{8.14}$$

Measuring auditability for offered cloud services

Regarding the presented trust tree and its related explanations (figure 4-1 in chapter 4), we can consider that auditability will be measured by measuring the previous characteristics. Subsequently, auditability of a cloud service provider can be measured by equations 8.15-8.17 which are the results of equations related to the categories of individual (equations 8.1, 8.4, 8.7 ,8.12 ,8.13 and 8.14), organization (equations 8.2, 8.5, 8.8, 8.10, 8.12, 8.13 and 8.14) and cloud service provider (equations 8.3, 8.6, 8.9, 8.11, 8.12, 8.13 and 8.14) respectively.

Auditability with measurable measures for individuals :

$$E_{A_i} = E_{P_i} + E_{S_i} + E_{R_i} + E_U + E_I \tag{8.15}$$

Auditability with measurable measures for organization :

$$E_{A_o} = E_{P_o} + E_{S_o} + E_{R_o} + E_{F_o} + E_U + E_C + E_I \qquad (8.16)$$

Auditability with measurable measures for cloud service provider :

$$E_{A_c} = E_{P_c} + E_{S_c} + E_{R_c} + E_{F_c} + E_U + E_C + E_I \qquad (8.17)$$

Phase 3. Calculation of the distance

In this phase of ED-BeCT, the values of the certain characteristics negotiated in SLAs of cloud providers in phase 1 and the results of measuring these characteristics obtained in phase 2, need to be considered. Then, by applying the equation (8.18), the distance between the ideal values and the measured values in previous phase for these trust characteristics will be calculated.

$$D_{C_i} = \left| x_{id_i} - x_{m_i} \right| \qquad (8.18)$$

Where D_{C_i} represents the distance between i-th ideal value x_{id_i} with the i-th measured value x_{m_i} in phase 2, for the (common) characteristic C in the SLAs , i = 1,…,n (n is the number of (common) characteristics that are mentioned in SLAs with their values).

Phase 4. Supplier filtration

In this phase regarding the results of phase 3, cloud user would be able to distinguish the trustable cloud service providers based on the following criteria :

- The least calculated values for distance (D_{C_i}) will be the desired values. Thus, the cloud providers that have the least distance between the values in their related SLAs and the measured values in phase 2, can be filtered to be evaluated in greater details in the next phases of ED-BeCT;
- The greater number of characteristics that are met in the SLA, can create the greater cloud provider trasparency for cloud users. Thus, this can be the other criterion to filter trustable cloud service providers.

The result of this phase would be certain number of cloud service providers that they have greater transparency and the characteristics in their SLAs have more accurate values (for which D_{C_i} has the lowest values). Thus, they are filtered as more trustable providers compared with the ones that are not filtered. Therein after, based on the category of cloud user (individual or organization or CSP) and the related figures in chapter 7 (figures 7-1, 7-2 and 7-3), the trust characteristics will be completely measured to assess the offered cloud services more accurately. The details of this assessment is presented in the next phases of ED-BeCT in which the focus is only on the filtered providers.

Phase 5. Measuring rest of the trust characteristics

In this phase, the remaining trust characteristics according to the related models (figures 7-1, 7-2 and 7-3 in chapter 7) will be measured for the filtered providers by applying the evaluation functions presented in phase 2.

Phase 6. Normalizing the calculated values

In the present model, the equation (8.19) is applied to normalize the calculated values in phase 5 to calculate the level of trust for the filtered cloud service providers in theprevious phase [263]. Since there is no non-beneficial characteristic (the characteristic that its preferable

values are minima) [263], we use equation (8.19) for beneficial characteristics (the ones that their preferable values are maxima).

$$\bar{x}_{ij} = \frac{x_{ij} - min_{i=1}^{m}(x_{ij})}{max_{i=1}^{m}(x_{ij}) - min_{i=1}^{m}(x_{ij})} \quad (8.19)$$

(Adapted from [263])

In this equation, x_{ij} (i = 1,..m and j = 1,..n) is the measure of i-th cloud service provider regarding the j-th characteristic, m is the number of selected cloud service providers and n is the number of trust characteristics.

Phase 7. Calculating the weights for the characteristics

Obviously, each of the cloud service users' requirements may differ according to their defined goals. As already explained, the presented trust characteristics are the users' requirements' representations. Thus, this diversity in requirements has direct effect on the importance of trust characteristics for each cloud user. Hence, to evaluate the trustworthiness of the cloud service provider as accurate as possible, the level of importance of each trust characteristic should be taken into consideration.

On the other hand, based on the discussions in chapters 3 and 6 and according to the quality standards, sub-characteristics have a pivotal role in assessing trust characteristics in cloud computing. Thus they have the same level of importance which is = 1. In addition, based on the fact that in indirect measurement (which includes measuring characteristics such as performance efficiency, reliablity, etc.) the characteristics are unmeasurable quality characteristics and to make them measurable, they decomposed into various sub-characteristics [203], each sub-characteristic can be measured by their related measures (however, some of them need further research to be measurable by the identified measures in cloud environments such as buffer overflow). All these measures have the same role in assessing the related sub-

characteristic. Further, according to the results of chapter 7, the presented measures in the category of « doable » are fundamental to assess trust characteristics and they have a determining role to assess each characteristic for the offered cloud services. In addition, since the main goal of this research project is to evaluate different aspects of trust, considering all these measures in trust assessment equally, can be helpful in this regard. Therefore, to measure the level of each characteristic the pure calculation of these measures would create more credible results. Accordingly, we consider the same level of importance (= 1) for the measures as well.

In the present model, by applying Full Consistency Method (FUCOM) (taken from [137]), which is a novel multi-criteria problem solving method, the weights of each characteristic can be calculated based on users' demands.

The weight calculation will be performed in three steps (taken from [137]):

Step (1) - In the first step, the trust characteristics (C) are ranked according to their importance level for the cloud service user. This ranking will start from the characteristic with the highest importance to the trust characteristic of the least importance. Hence, the characteristics are ranked based on the expected values of the weights that can be obtained.
(A sample ranking as an example is illustrated in (8.20)):

$$C_{i(1)} > C_{i(2)} > \ldots > C_{i(k)} \tag{8.20}$$

In (8.20), i = 1,…,n (n is the number of selected trust characteristics) and k represents the rank of the characteristic and if there are two or more characteristics with the same level of importance, instead of (>), (=) will be placed.

Step (2) – In this step, the comparative priority ($P_{k/(k+1)}$, k = 1, 2, 3,..., n, where k represents the rank of characteristic) of the characteristics is determined. The comparative priority of the characteristics ($P_{k/(k+1)}$) is an importance of the $C_{i(k)}$ rank compared to the characteristic of

the $C_{i(k+1)}$ rank. Therefore, the vectors of the comparative priorities of the ten proposed trust characteristics in figure 3-4 in chapter 3 (we consider this figure, as the reference figure), can be determined as in (8.21):

$$V = (P_{1/2}, P_{2/3}, \ldots, P_{9/10}) \tag{8.21}$$

The comparative priorities of the characteristics are defined according to a predefined scale (i.e. the scale of [1,9]). This comparison of the trust characteristics is done regarding the first-ranked (the most important) characteristic. Hence, the importance of the characteristics ($w_{Ci(k)}$) for all of the characteristics ranked in step (1) of FUCOM is determined. The first-ranked characteristic is compared with itself (its importance is $w_{C1(1)} = 1$). Regarding this fact, the 9 comparisons of the characteristics need to be performed (the vectors of the comparative priorities in (8.21)).

Step (3) – This step is related to the calculation of the final values of the weights of the selected trust characteristics $(w_1, \ldots, w_{10})^T$. There are two conditions that should be satisfied by the final values of the weights.

Condition 1: The ratio of the weights should be equal to the comparative priority defined in step (2) of FUCOM. Therefore, the following condition is met;

$$\frac{w_k}{w_{K+1}} = P_{k/(k+1)} \tag{8.22}$$

Condition 2: The condition of mathematical transitivity that is;

$$P_{k/(k+1)} \otimes P_{(k+1)/(k+2)} = P_{k/(k+2)}$$

should be satisfied by the final values of the weights. Since $P_{k/(k+1)} = \frac{w_k}{w_{K+1}}$ and $P_{(k+1)/(k+2)} = \frac{w_{k+1}}{w_{K+2}}$, thus $\frac{w_k}{w_{K+1}} \otimes \frac{w_{k+1}}{w_{K+2}} = \frac{w_k}{w_{K+2}}$. Regarding these explanations, another condition that the final values of the weights of the characteristics need to meet is:

$$\frac{w_k}{w_{K+2}} = P_{k/(k+1)} \otimes P_{(k+1)/(k+2)} \tag{8.23}$$

Full consistency i.e. minimum deviation from full consistency or DFC (X) of the comparison, will be satisfied only if transitivity is fully respected. It means that when the conditions of (8.22) and (8.23) are met. Thus, the requirement for maximum consistency will be fulfilled; which is DFC equal to zero ($X = 0$) for the calculated weights. Therefore, it is necessary that the values of the weights $(w_1,\ldots,w_n)^T$ (n is the number of selected trust characteristics) meet the condition of $\left|\frac{w_k}{w_{k+1}} - P_{\frac{k}{k+1}}\right| \leq X$ and $\left|\frac{w_k}{w_{k+2}} - P_{\frac{k}{k+1}} \otimes P_{\frac{k+1}{k+2}}\right| \leq X$, with the minimization of the value X. In this way, the requirement of maximum consistency will be satisfied. Hence, the final model to calculate the final values of the weights of the proposed trust characteristics in figure 3-4 in chapter 3, is presented in (8.24): (n is the number of selected trust characteristics).

$$\min X$$

$$s.t. \begin{cases} \left|\frac{w_{i(k)}}{w_{i(k+1)}} - P_{\frac{k}{k+1}}\right| \leq X & \text{for (i = 1, ..,n)} \\ \left|\frac{w_{i(k)}}{w_{i(k+2)}} - P_{\frac{k}{k+1}} \otimes P_{\frac{k+1}{k+2}}\right| \leq X & \text{for (i = 1, ..,n)} \\ \sum_{i=1}^{n} w_i = 1 & \text{for (i = 1, ..,n)} \\ w_i \geq 0 & \text{for (i = 1, ..,n)} \end{cases} \tag{8.24}$$

Phase 8. Calculating the level of trustworthiness

Once the weights of characteristics were calculated to represent their importance and influence on the evaluation results [264], they need to be applied in the normalized values calculated in phase 6. By using SAW (Simple Additive Weighting) method with the presented equation (8.25), the level of trustworthiness of each filtered cloud service provider in phase 4 will be obtained:

$$T_i = \sum_{j=1}^{n} \bar{x}_{ij}\, w_j \qquad (8.25)$$

Where T_i is the level of trustworthiness of i-th cloud service provider regarding the j-th characteristic, $\bar{x}_{ij}$ is the normalized measure of i-th cloud service provider regarding the j-th characteristic, w_j is the calculated weights of the characteristics in phase 3 and n is the number of trust characteristics that in the presented model in figure 3-4 in chapter 3, there are 10 proposed trust characteristics.

One advantage of the SAW method in the proposed model is that the weights in the equation (8.25), have a great influence on the final result [264]. Sinc, the requirements of each cloud service user is different, the level of importance of each trust characteristic by using Full Consistency Method (FUCOM) is taken into consideration. In addition, in FUCOM the maximum consistency and minimum deviation from full consistency to calculate the weights of characteristics is fulfilled. This helps to preserve the accuracy of the calculations.

Therefore, to calculate the level of trustworthiness in cloud environments dynamically and according to the users' requirements, the combination of SAW method and FUCOM can be the best possible solutions among existing multi-criteria decision methods (MCDM) in the time of conducting this research.

Phase 9. Comparing the results and making decision

According to the results obtained in the previous phase, the highest level of trustworthiness will be awarded to a cloud service provider having the highest T_i value. Thus, this cloud service provider would be the best choice among the other filtered cloud service providers.

Figure 8-1 summarizes the phases of ED-BeCT.

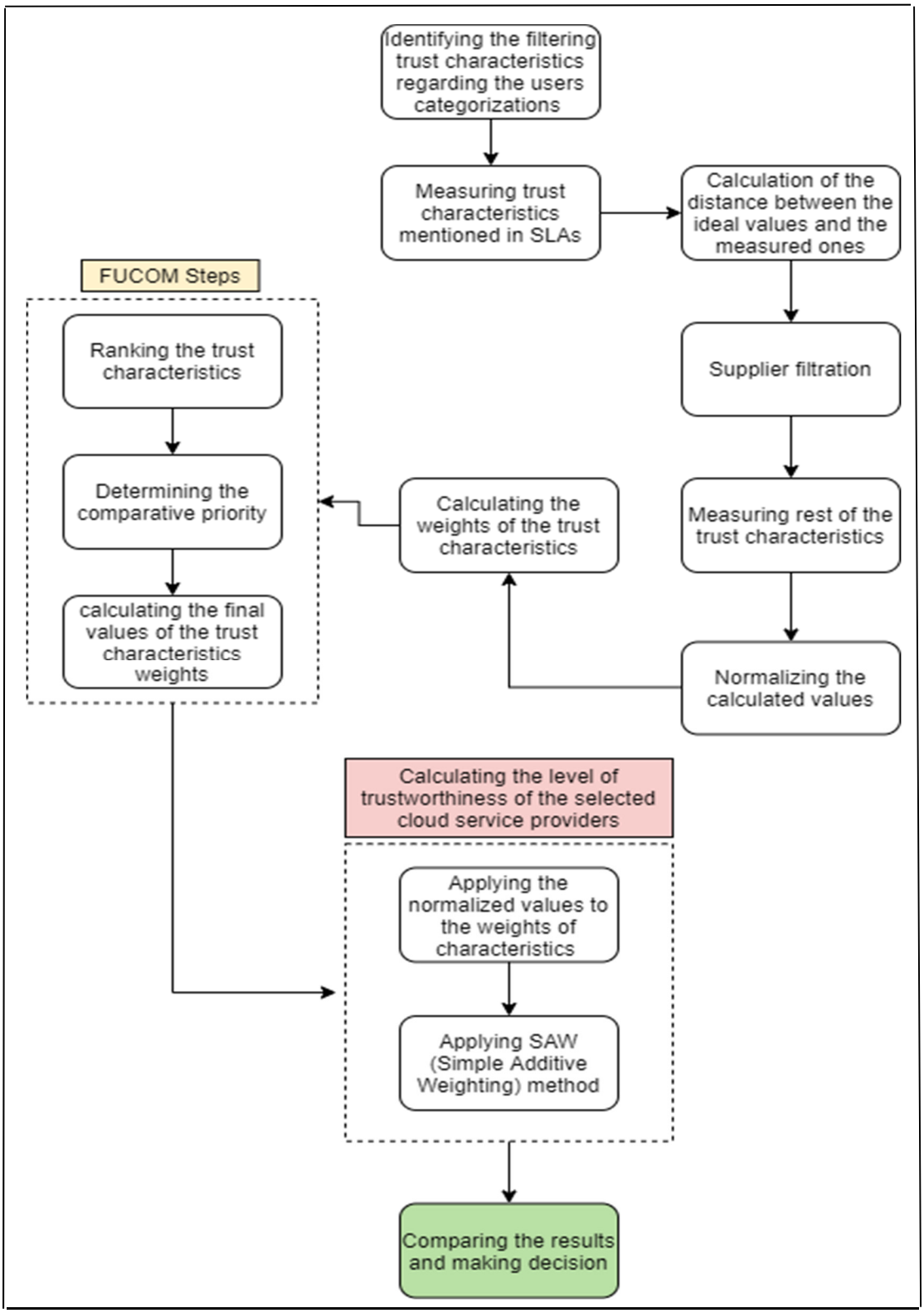

Figure 8-1. The phases in ED-BeCT

CHAPTER 9

CASE STUDY

In this chapter, an example to show the applicability of the proposed cloud trust model (ED-BeCT) is designed to achieve a better understanding. This example is tailored in the domain of medical industry from individual perspective in the categories of cloud users explained in chapter 7 to select a trustworthy cloud service provider for IaaS services.

In this example by considering the three business goals of the user that are explained in the first phase of ED-BeCT and considering four related characteristics to summarize the calculations, we illustrate the phases of this model. However, in the reality it is recommended to consider all the trust characteristics presented in chapter 7 for each category of cloud user, as users have a great variety of requirements and these characteristics can evaluate the behavior of the cloud services based on the different aspects of trust (from quality of services to some intrinsic characteristics of cloud computing explained in chapter 7).

9.1 Example development

To provide quality care to patients, health providers significantly depend on medical imaging devices, i.e. ultrasound, magnetic resonance, positron emission tomography, computed tomography, endoscopy and computed radiography [265]. These medical devices normally produce a huge amount of data that deposit remarkable burden on hospital computing, storage and network infrastructures [265].

Picture archiving and communication systems (PACS) are broadly used in healthcare domains to provide cost-effective storage of, and easy access to, the huge amount of medical images from multiple modalities such as CT, MRI, radiography, etc. [265]. In addition, these systems are used to systematically process the acquired images to diagnose the patient's medical condition in a very short time [266]. In PACS the medical images are stored in Digital Image and Communications in medicine (DICOM) file format and transferred to the workstations

where they can be accessed [265]. Non-image files may be unified by the consumer industry formats like PDF [265].

PACS vendors tend to provide scalable systems with high-level disaster recovery provisions and flexible configurations while providing economical hardware and software support [267]. This can be achievable by migrating data from a traditional onsite archiving system to the cloud environments [267]. Subsequently, two major components of a common PACS that can be outsourced to the cloud are [268] :

- The DICOM object repository, that normally requires an infrastructure with huge storage capacity;
- The database system, that is typically a relational database management system (RDBMS) that supports the DICOM Information Model (DIM) and comprises mandatory metadata pertinent to the patients, studies, series and images.

> « When a medical image archive receives studies from imaging modality equipment, it is necessary to store the images in the file system repository and to update the database with data elements extracted from the received study headers. [268] »

Accordingly, a PACS vendor enterprise aims to use infrastructure as a service (IaaS) provided by a cloud service provider to deploy the two components of this medical software and share the medical information among hospitals and clinics with the goal of accessing data anywhere and anytime via cloud environments. In addition, the PACS vendor enterprise expects the IaaS provider to [190] :

- Deploy the PACS web base applications to prepare infrastructure for the hospitals and medical centers;
- Provide load balancing services;
- Facilitate the process of PACS application propagations on infrastructure instances;
- Make the infrastructure services available for the hospitals and medical centers;

- Preserve the security of CPUs, data and network;
- Manage account and provisioning.

In this case, PACS vendor is an enterprise that is considered as an individual in our categorizations of cloud service users of which the hospitals, clinics and medical centers are the potential customers.

9.2 Applying the phases of ED-BeCT

Phase 1 : Identification of trust characteristics for CSP candidating

In this example, the PACS vendor enterprise is considered as the end user of the cloud services. Thus, in the presented categorizations of cloud users in chapter 6, its level of access to the cloud resources can be specified in the category of individual.

In this phase, first it is essential to specify the business goals for using cloud services. In this example, to summarize the process of ED-BeCT for better understanding, imagine the PACS vendor enterprise will specify the three main business goals for using IaaS services that are [269] :

- Shifting medical data in cloud to store them in a secure and reliable way;
- Providing a system accessible anywhere and anytime;
- Providing a system that automatically adjusts the resources to the workload received from hospitals and medical centers by the application of PACS without human intervention.

It should be noted that as explained in previous chapter (in phase 3 of ED-BeCT), the sub-characteristics along with their measures have the same level of importance. Thus, in case of choosing each trust characteristic, all its presented sub-characteristics in the related model (in this example, figure 7-1 in chapter 1) should be considered. On the other hand, if one (or more) sub-characteristic(s) is (are) considered, to make the evaluation more precisely, it is

recommended to assess the other sub-characteristics in the related group. In addition, each sub-characteristic has some identified measures that they need to be considered with the same level of importance to evaluate the related sub-characteristic as accurately as possible.

In this example, since we consider only three types of requirements, thus it is necessary to link them to their related trust characteristics in figure 7-1 in chapter 7 for the category of individual (the PACS vendor enterprise). This correlation is justified as follows :

- The first requirement – Obviously, security and reliability of the cloud services are the two related trust characteristics. As explained, the PACS vendor enterprise has many customers (hospitals and medical centers). Thus, there exist several patients for which there are diverse data and confidential information that are supposed to be stored in cloud storage. Consequently, security of the cloud services will be one of the fundamental trust characteristics;

  In addition, based on the performed analysis in section 6.3 of chapter 6 and regarding the sub-characteristics of reliability (availability and continuity) and their measurable measures for the category of individual, reliability is also an essential trust characteristic for the PACS vendor enterprise. Since several hospitals and medical centers are its customers, each of these customers will have sensitive data which will be in cloud environment and they must be available whenever required under all foreseeable circumstances (continuity) through cloud services. Thus, using a reliable cloud service can have direct influence on the applicability of PACS instances in cloud and increase the customer satisfaction (hospitals);

- The second requirement – Providing a system accessible anywhere and anytime through cloud environment in particular a medical system that it is very essential to be usable in emergency situations through cloud services, the necessity of usability as one of the trust characteristics can justify. Thus, if the usability of the selected cloud service is weak, it can impact on the PACS instances in cloud environments which can decrease the customers

satisfaction level of the PACS vendor enterprise significantly. Consequently, the PACS vendor enterprise will undergo a huge damage in its market;

- The third requirement – This requirement to some extent defines the effects of load balancing and auto-scaling in cloud computing which are supposed to be provided by CSPs. To specify the related trust characteristic to this requirement, it is essential to clarify the concepts of load balancing and auto-scaling;

- **Load balancing** : To enhance the overall performance of the system, it is necessary to distribute a dynamic large local workload between all the nodes in cloud environment by the techniques of load balancing [270] :

- To allocate the computing resources fairly and achieve the customers of the PACS application satisfaction (in this case, the hospitals and medical centers) and accurate resource utilization to minimize resource consumption, and also implement fail over and scalability (performance efficiency), load balancing would be substantial;
- Load balancing by dividing the traffic between all servers, helps to eliminate delays in sending and receiving data and can provide a maximum throughput with minimum response time (performance efficiency);
- There are various algorithms to implement load balancing in cloud computing (such as random, round-robin, etc.) by the cloud service providers. Thus, it is necessary to assess this capability of the proposed cloud services to ensure their behavior will be compatible with the requirements of this enterprise. As the PACS vendor enterprise could not be able to have access to the execution process of the services and also the policy of implementation in cloud environments would not be transparent, since they are confidential information of cloud service providers, thus, the measures of time behavior (sub-characteristic of performance efficiency) related to the provided cloud services can be a helpful criterion;

Generally speaking, all the incoming requests will be received and forwarded by the load balancer to one of the servers in the pool, which can be implemented in various ways [269].

In any case, the important point is that the load balancer has the latest information about the active virtual machines (VMs) to stop sending requests to removed VMs and to start sending workload to the recently added VMs [269].

- **Auto-scaling** : As explained in ISO/IEC 17788 [24], one of the essential characteristics of cloud computing is elasticity. Elasticity allows the applications to provision (acquire) and de-provision (release) of the resources dynamically and adjust to changing demands [269]. But, making decision for the exact amount of resources in cloud environments can be challenging [269];

  Given that the main business goal of PACS vendor for using IaaS services is shifting medical data in cloud, to provide a system that automatically adjusts the resources to the workload received from hospitals and medical centers by the application of PACS without human intervention [269], evaluating the capability of auto-scaling provided by the IaaS providers would be crucial for this enterprise, because of following reasons [269] :

- To have a remarkable reduction in the cost of the storage for the hospitals, and also the cost of using resources in cloud environments, it is important for this enterprise that the auto-scaler be aware to make the cost-effective decisions. This would be an attractive reason to increase the sale of PACS systems and a huge success for the PACS vendor enterprise.
- The auto-scaler is responsible to maintain an acceptable Quality of Service (QoS) to secure the correct functionality of the application. As explained in [269], (in the current case) the QoS rely on two types of Service Level Agreement (SLA): a) the application SLA, that is a contract between application owner (the PACS vendor enterprise) and its potential customers (hospitals and medical centers), b) the resource SLA, which is a contract between the cloud service provider and the application owner (the PACS vendor enterprise). Basically, both types of SLA need to be combined to meet the different requirements in application and resources.

Basically, the auto-scaling process comprises four phases [269] :

- Monitoring: the auto-scaling system requires the information about the system and application state, gathered by a monitoring system;
- Analyzing: based on the information obtained in monitoring phase, in this phase the auto-scaler performs an analysis about current system utilization or required resources and optionally predicts the future requirements of the system. Accordingly, there are two types of auto-scaling, 1- reactive, that is without performing any kind of prediction, the current status of the system will be responded, 2- proactive, is a technique to predict the future needs to arrange resource allocation with adequate anticipation. This anticipation would be crucial for the PACS vendor enterprise due to the delay between auto-scaling execution (such as adding a server) and its effectiveness (i.e. the time that takes to assign a physical server to a VM, or move the VM image to it, deploy the operating system and application and make the server completely operational [269, 271]). Reactive systems are not able to deal with sudden traffic burst, whereas, proactivity may be helpful to handle fluctuating needs and scale in advance. This can reduce the response time significantly and increase the performance of the system that would be tangible for the users;
- Planning: it is related to the modality of scaling of the resources assigned to the application after determining the current (or future) state, such as removing a VM or adding memory to an especial VM;
- Executing: the actual execution of scaling decided in previous phase will be done through the cloud provider's API in this step. The actual complexities are not transparent to the PACS vendor enterprise, but the time that takes from requesting a resource until it is available, would be important and may be part of the resource SLA.

Although, the policies and deployed algorithms of load balancing and auto-scaling are not transparent for the PACS vendor enterprise, they have direct influence on the performance of provided cloud services. Thus based on the aforementioned explanations, assessing performance efficiency by considering all its sub-characteristics along with their presented measures can provide an insight to ensure the compatibility of the identified related requirements with the offered cloud services.

Therefore, the four trust characteristics of performance efficiency, usability, security and reliability can be specified related to the three explained requirements of the PACS vendor enterprise. Then, the PACS vendor enterprise should negotiate with the cloud service provider to identify the values of these characteristics in the proposed service level agreement (SLA). Based on the proposed SLA, the common characteristics among 4 identified characteristics in this example, should be specified between SLAs. The values of these characteristics in the related SLA are considered as ideal trust characteristics and since they are negotiated characteristics, they are supposed to have the same values in the SLA of each CSP. In this phase, the PACS vendor enterprise regarding the reputation of the cloud providers and by taking into account the feedback and recommendations of the other users, can limit the number of providers to negotiate.

Subsequently, the result of phase 1 of ED-BeCT, will be one or more trust characteristics (among the 4 identified ones) that their values in the SLAs related to each CSP are considered as ideal values. In this example, imagine there is availability (one of the sub-characteristics of reliability) as the common element in the negotiated SLA (the value of availability in each SLA may differ) which is the ideal sub-characteristic.

Phase 2 – Measuring (common) trust characteristics mentioned in SLAs

In this phase, the PACS vendor enterprse by applying the evaluation functions in table 9-1 which are extracted from system and software quality standards and discussed in previous chapter, and using the equation (9.1) would be able to measure availability related to the cloud services of the cloud providers with which the PACS vendor enterprise had negotiations.

As presented in ISO/IEC 25010, ISO/IEC 25011 and ISO/IEC 25023, availability is sub-characteristic of reliability which can be measured by considering system availability and mean down time (ISO/IEC 25023).

Table 9-1. Availability measures

Measures (adapted from ISO/IEC 25023) [160]	Evaluation Function (adapted from ISO/IEC 25023) [160]
System availability	m_1 = T/t T: Provided system operation time t: Specified system operation time
Mean down time	m_2 = T/b T: Total down time b: The number of observed breakdowns

$$\text{Availability} = \sum_{i=1}^{2} m_i \tag{9.1}$$

Imagine, there are 10 cloud service providers with which the PACS vendor enterprise had negotiation. The PACS vendor enterprise needs to calculate the value of availability by the help of table 9-1 and equation (9.1), for each of these 10 cloud service providers separately.

Phase 3 – Calculation of the distance between the ideal values of trust characteristics and the measured ones

In this phase of ED-BeCT, the values of the availability negotiated in SLAs of cloud providers in phase 1 and the results of measurements obtained in phase 2, need to be considered. Then, by applying the equation (9.2), the distance between the ideal values and the measured values for availability of cloud services for each provider will be identified.

$$D_{C_1} = \left| x_{id_1} - x_{m_1} \right| \tag{9.2}$$

Where D_{C_1} represents the distance between the ideal value x_{id_1} in SLA with the measured value x_{m_1} in phase 2, for availability which is common in the SLAs. This calculation will be applied for each of these 10 cloud service providers separately.

Phase 4 – Supplier filtration

In this phase, regarding the results of phase 3, the PACS vendor enterprise is able to distinguish the trustable cloud service providers. Imagine, among 10 CSPs with which the PACS vendor enterprise had negotiations, four providers (Supplier A, Supplier B, Supplier C and Supplier D) have been filtered based on the following criteria:

- These 4 providers have the least calculated values for distance (D_{C_1}) between the ideal values of availability in their SLAs and the measured values in phase 2;
- There is the greater number of characteristics that are met in their SLA. Thus, it can create the greater transparency of these cloud providers for the PACS vendor enterprise.

Therefore, these 4 CSPs are filtered as more trustable providers compared with the ones that are not filtered. Therein after, based on the category of the PACS vendor enterprise (individual) and its related model (in chapter 7 figures 7-1), the trust characteristics will be completely measured for these 4 CSPs to assess the offered cloud services more accurately. The details of this assessment is presented in the next phases of ED-BeCT which are only focused on the 4 filtered providers.

Phase 5 – Measuring rest of the trust characteristics

In this phase, the remaining number of trust characteristics according to the related model for the category of individual in figures 7-1 in chapter 7 will be measured for the 4 filtered providers by applying the evaluation functions presented in phase 2. (As in this example we only focus on the four characteristics of performance efficiency, usability, security and reliability, we perform the next phases of ED-BeCT with these characteristics.)

Regarding the presented model for individual category of cloud users and the explanations in chapter 7, performance efficiency, usability, security and reliability can be measured by the

PACS vendor enterprise for Supplier A, Supplier B, Supplier C and Supplier D by applying following equations from chapter 8:

Performance efficiency : $$E_{P_i} = \sum_{j=1}^{5} m_j \quad (9.3)$$

Usability : $$E_U = \sum_{j=1}^{24} m_j \quad (9.4)$$

Security : $$E_{S_i} = \sum_{j=5}^{9} m_j \quad (9.5)$$

Reliability : $$E_{R_i} = \sum_{j=5}^{6} m_j + m_{12} \quad (9.6)$$

Table 9-2 shows the final results of these four trust characteristics in this example. It should be noted that in this case study these evaluations are done hypothetically.

Table 9-2. The results of trust characteristics evaluation

	Performance Efficiency (C_1)	**Security (C_2)**	**Usability (C_3)**	**Reliability (C_4)**
Supplier A	122	102	58	96
Supplier B	121	100	85	111
Supplier C	98	145	110	78
Supplier D	165	150	89	63

Phase 6 - Normalizing the calculated values

As explained in chapter 8, we use equation (9.7) for the selected beneficial characteristics (the ones that their preferable values are maxima). Table 9-3 illustrates the normalized values.

$$\bar{x}_{\text{ij}} = \frac{x_{ij} - min_{i=1}^{m}(x_{ij})}{max_{i=1}^{m}(x_{ij}) - min_{i=1}^{m}(x_{ij})} \quad (9.7)$$

Where x_{ij} (i = 1,..4 and j = 1,..4) is the measure of i-th cloud service provider regarding the j-th characteristic, m is the number of selected cloud service providers and n is the number of trust characteristics. Thus, in this scenario, we have 4 trust characteristics and 4 filtered cloud service providers.

Table 9-3. The normalized values

	Performance Efficiency (C_1)	**Security (C_2)**	**Usability (C_3)**	**Reliability (C_4)**
Supplier A	1	0.687	0	0.594
Supplier B	1	0.417	0	0.722
Supplier C	0.299	1	0.478	0
Supplier D	1	0.853	0.255	0

Phase 7 - Calculating the weights for the characteristics

As explained in chapter 8, by applying Full Consistency Method (FUCOM), the PACS vendor enterprise can calculate the weights of each characteristic based on its demands. The calculation will be done in three steps:

Step (1) – The first step is ranking the characteristics: $C_2 > C_1 > C_4 > C_3$

Step (2) – In the second step, the PACS vendor enterprise will perform the pairwise comparison of the ranked characteristics from step (1). This comparison will be done based on the scale [1,9] and regarding the first-ranked characteristic which in this example is C_2. Hence, the priorities of the characteristics ($W_{Ci(k)}$) for all the three ranked characteristics in the previous step were determined. In table 9-4, the priorities of these four characteristics ($\in$[1,9]), are presented.

Table 9-4. Priorities of the characteristics

Characteristics	$\mathbf{C_2}$	$\mathbf{C_1}$	$\mathbf{C_4}$	$\mathbf{C_3}$
$\mathbf{w_{Ci(k)} \in [1, 9]}$	1	4.5	6	7

According to the determined priorities (table 9-4) and regarding the condition $\frac{w_k}{w_{K+1}} = P_{k/(k+1)}$, the comparative priorities of the characteristics can be calculated:

P_{C_2/C_1} = 4.5/1 = 4.5, P_{C_1/C_4} = 8/4.5 = 1.33, P_{C_4/C_3} = 7/6 = 1.17.

Step (3) – In the last step, the final values of the weights need to meet two conditions.

Condition 1: $\frac{w_k}{w_{K+1}} = P_{k/(k+1)} \rightarrow \frac{w_2}{w_1} = 4.5, \frac{w_1}{w_4} = 1.33, \frac{w_4}{w_3} = 1.17$

Condition 2: $P_{k/(k+1)} = \frac{w_k}{w_{K+1}}$ & $P_{(k+1)/(k+2)} = \frac{w_{k+1}}{w_{K+2}} \rightarrow \frac{w_k}{w_{K+2}} = P_{k/(k+1)} \otimes P_{(k+1)/(k+2)}$

thus, $\frac{w_2}{w_4} = \frac{w_2}{w_1} \otimes \frac{w_1}{w_4} = 4.5 \times 1.33 = 5.99$, $\frac{w_1}{w_3} = \frac{w_1}{w_4} \otimes \frac{w_4}{w_3} = 1.33 \times 1.17 = 1.56$.

To calculate the final values of the weights, the following equation is applied:

$$\min X$$
$$s.t. \begin{cases} \left|\frac{w_2}{w_1} - 4.5\right| \leq X \ , \left|\frac{w_1}{w_4} - 1.33\right| \leq X \ , \ \left|\frac{w_4}{w_3} - 1.17\right| \leq X \ , \\ \left|\frac{w_2}{w_4} - 5.99\right| \leq X \ , \ \left|\frac{w_1}{w_3} - 1.56\right| \leq X \\ \sum_{i=1}^{4} w_i = 1 \\ w_i \geq 0 \qquad \text{for (i = 1, ..,4)} \end{cases} \tag{9.8}$$

Subsequently, by solving this equation, the final values of the weights for the characteristics $(0.145,0.653,0.093,0.109)^{T}$ and DFC of results $X = 0$ are obtained.

Phase 8 – Calculating the level of trustworthiness of the selected cloud service provider

The calculated weights of characteristics in previous phase which represent the level of importance of the selected characteristics and their influence on the evaluation results [264], have to be applied in the normalized values, calculated in phase 6. By using SAW (Simple Additive Weighting) method, the level of trustworthiness of each filtered cloud service provider in phase 4, will be obtained using the following equation:

$$T_i = \sum_{j=1}^{n} \bar{x}_{ij}\, w_j \tag{9.9}$$

Where T_i is the level of trustworthiness of i-th cloud service provider regarding the j-th characteristic, $\bar{x}_{ij}$ is the normalized measure of i-th cloud service provider regarding the j-th characteristic, w_j is the calculated weights of the characteristics in phase 7 and n is the number of trust characteristics that in this example there are 4 trust characteristics.

The levels of trustworthiness of the selected cloud service providers are calculated as follows:

T(Supplier A) = 0.145 + 0.687×0.653 + 0 + 0.594×0.109 = 0.658

T(Supplier B) = 0.145 + 0.417×0.653 + 0 + 0.722×0.109 = 0.496

T(Supplier C) = 0.299×0.145 + 0.653 + 0.478×0.093 + 0 = 0.741

T(Supplier D) = 0.145 + 0.853×0.653 + 0.255×0.093 + 0 = 0.726

Phase 9 – Comparing the results and making decision

According to the results obtained in the previous phase, the highest level of trustworthiness will be awarded to a cloud service provider having the highest T_i (the level of trustworthiness of i-th cloud service provider) value which in this example is Supplier C. Thus, this cloud service provider would be the best choice among the other filtered cloud service providers. Figure 9-1 illustrates the level of trustworthiness calculated for these 4 filtered cloud service providers. According to this figure, Supplier D can be the second option, Supplier A can be the third option and the last option can be Supplier B for this example. This ranking would be helpful in case of having some challenges with Supplier C such as cost of the services.

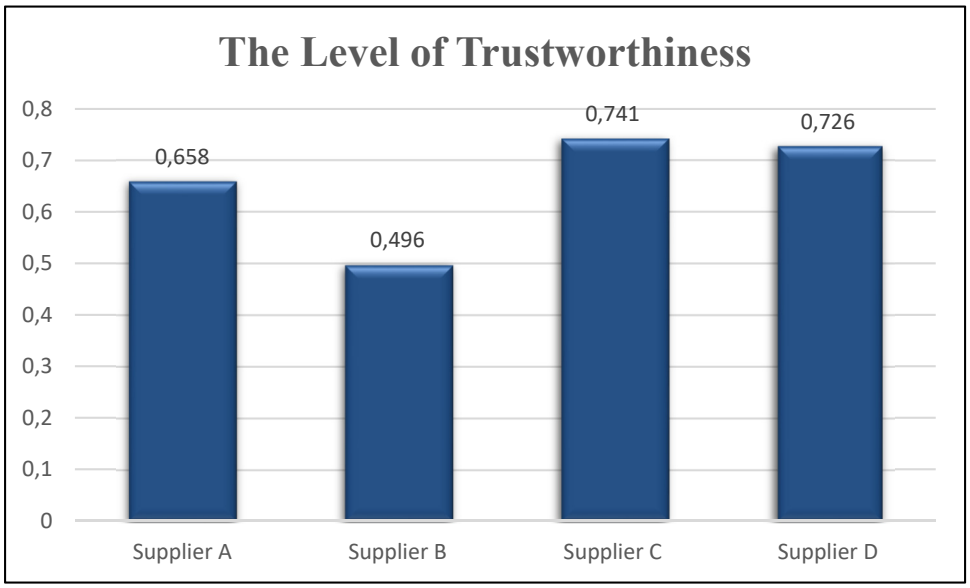

Figure 9-1. The level of trustworthiness calculated for 4 filtered cloud service providers

CHAPTER 10

SUMMARY OF RESEARCH ACHIEVEMENTS, OBTAINED RESULTS AND THE POTENTIAL USE

Cloud computing has emerged as a powerful technology to benefit from on-demand computer system resources without users' interventions directly. Althought, the intention of this technology is captivating as its main target is to reduce the cost of the IT department and make the IT-based resouces always available, there are still various issues and challenges associated with applying this technology which are not solved yet.

A large and growing body of the literature has investigated the role of cloud computing in facilitating data management, sharing hardware and software resources and having a protected interaction between cloud service users and cloud service providers. Therefore, we can say that there is a bilateral relationship between a cloud user and a cloud provider that requires a consensus-based standard or a agreed rule in which various influential factors of cloud computing are defined as references. These references can be considered for the purpose of cloud service evaluation. However, researchers in previous studies have not treated this issue in much detail.

On the other hand, based on the different requirements and challenges of cloud computing investigated in this research project and explained in chapters 1, 3 and 4, we demonstrated the relationship of trust evaluation in cloud computing with selection of a trustworthy cloud service provider. Much of the current literature on trust and trustworthiness pay particular attention to evaluate the level of trust in cloud environments. This evaluation is principally done based on some selected characteristics that mainly have direct relationships with users' requirements. However, there is not any criteria or consensus-based standard to designate these characteristics precisely. Accordingly, in this thesis we proposed a cloud trust model to overcome the challenges of using cloud computing technology and address the limitations of the previous works in this area.

In this chapter, the main achievements of this study and the potential usages from which the industry will profit are summarized. In addition, the results and findings of the previous chapters are gathered and briefly explained.

10.1 Summary of the results

In this section, we summarized the main results of this research project as follows :

- Results of the literature review – Details in chapter 1

o The first purpose of this study is to find the main obstacles for using cloud computing technology. According to the literature, there are various issues and challenges in this technology to be fully accepted by the users. These challenges can be categorized into two groups :

- technical aspects : such as performance efficiency;
- psychological aspects : as managing cloud resources and data is with minimal users' intervention, from users' perspective some concerns such as data security and quality of services are hypercritical.

As already discussed in the previous chapters, the aforementioned challenges are addressed under assessing the trustworthiness of the cloud service provider.

o To reply to the first fundamental question that is « what » should be evaluated to enable cloud users based on which judge the cloud services, a literature review on trust characteristics was conducted to identify trust characteristics that were proposed in domain-related research papers. However, we found no consensus-based set of trust characteristics and there is no agreed rule related to the quality of cloud services. An implication of this is the possibility that the users may not be able to perform the trust assessment accurately

since a considerable part of this assessment would be based on the general knowledge rather than agreed standards;

- Based on the methods for categorizing trust models in the literature, we proposed « measure-based-behavioristic trust models » as a new category to accommodate the proposed model in this research project. The models under this category can combine different techniques to analyze the behavior of the provided cloud services based on the measures identified for the required characteristics. The important point in this category is « measures » of the characteristics;

- The 11 criteria which some of them are extracted from the literature and explained in chapter 1, are considered as the assessment criteria for the proposed category (measure-based-behavioristic trust models). These criteria have a noteworthy role in developing a comprehensive trust model. However, they are not addressed in the previous studies completely. In table 1-13, chapter 1 we provided the details of how we addressed these criteria in the current project.

- Results of comparing trust characteristics in the literature with the applicable standards – Details in chapter 3

- Based on the discussion in chapter 3, we found that there are many published trust characteristics grouped in various cloud trust models, however, they present the points of view of the researchers who developed them. Additionally, most of these trust models pay special attention to security, while without the other aspects of trust, the assessment of trust in the cloud concept would not be complete. Moreover, there are several standards related to cloud computing, however, the level of their completeness and maturity is not stable yet;

- We presented a brief overview of the applicable standards and described their different parts that can be related to the essential characteristics and features of cloud services. We conducted comparison analysis between the extracted characteristics from the literature in

chapter 1 and the characteristics defined in the investigated standards. Based on this comparison we found that there are several standards presenting software quality models (and characteristics and measures associated to them). These models represent high completeness and maturity, but their applicability to cloud computing technology has not been fully verified yet;

- According to the analyzed papers (chapter 1) and ISO/IEC standards (chapter 3) trust comprises many characteristics, but identified set of commonly recognized characteristics (table 3-5, chapter 3) can reflect the broad, close to a consensus view on main aspects of cloud computing. Hence, for assessing trustworthiness of a cloud service provider, these characteristics could play a significant role as the essential trust characteristics;

- The final result of this chapter is the minimal prototype model for cloud services trust. This model encompasses 10 characteristics, with some of them being further decomposed into sub-characteristics. These characteristics are identified based on analyzing the literature and applicable standards in cloud computing.

- Results of trust characteristics measures – Details in chapter 4

Basically, the trust characteristics presented in the literature, are related to the principal aspects of cloud computing. However, the measures we found in the literature for assessing the proposed trust characteristics do not conform to system and software quality standards, nor are they easily applied to the context of cloud computing. Since there is no rule to select cloud trust characteristics measures, we chose instead to refer to existing standards and identify these measures. We chose the investigated standards in chapter 4 as they are agreed-upon reference, available, and have scientific and academic support.

Chapter 4 aimed to determine the measures of trust characteristics based on analyzing standard characteristics of cloud computing and IT services, extracted from system and software quality standards and cloud computing standards. Figure 4-1 in chapter 4, illustrates the trust tree of

cloud computing along with the standard measures which can clearly clarify the dependencies and existing correlations among the proposed characteristics. In addition, this trust tree is complementary to the results of which are depicted in figure 3-4 in chapter 3.

The most interesting aspects of the presented trust tree are as follows :

- Auditability is placed as the sub-layer of trust which is the measure of trustworthiness in cloud computing since the evaluation of cloud services by cloud providers may not be done honestly. This can be the main problem in the cloud computing adoption. Therefore, outsourcing the auditing cloud services to a trusted third party can produce more trustworthy results. Taking into account this fact, it can thus be concluded that by outsourcing auditing procedures, users are searching for a way to boost their confidence about cloud service provider selection and to ensure that the selected cloud services can cover the determined requirements. In the light of these principles and regarding the definition of trust in the system/software quality standard (ISO/IEC 25010) the correlation between trust and auditability can clearly be justified;

- Auditability of the system can be assumed as a gateway in its evaluation. If this gateway does not exist in the system, there is a possibility that the system may be unavailable for any form of real, fact-based evaluation.

- Results of an investigation into IaaS, PaaS, SaaS – Details in chapter 5

Regarding the explanations of IaaS, PaaS and SaaS in chapter 5, the following three main conclusions are achieved :

- The main similarity of the three cloud service models is the abstract way of provisioning of the resources;

- The key difference between IaaS, PaaS and SaaS could be the access level of the cloud service user and the cloud service provider in terms of each cloud service model. Figure 5-3 in chapter 5 can clearly describe this difference. It is apparent from this figure that in terms of SaaS, the cloud users have the least control on the cloud resources. By contrast, it can be seen that data and applications are under control of the cloud users in PaaS as well as IaaS. But cloud users have the broadest level of access by using IaaS and in comparison with SaaS and PaaS;

- Evaluating trust characteristics based on their measures which are discussed in the chapter 6, would depend on the level of access in cloud environments which is one of the significant criteria in evaluating the trustworthiness of the cloud service provider and measuring cloud services.

- Related results of chapter 5 and chapter 6 – Details in chapter 7

Based on the level of access to the cloud resources, we particularized the proposed set of trust characteristics in cloud computing (figure 3-4 in chapter 3) for the three specified categories of cloud service users. Thereby, individuals, organizations and also the owners of the cloud services (providers), can have practical measurements according to their related models. These models are explained in chapter 7.

The goals of presenting these models are explained as follows :

- These models provide a more accurate insight into the importance of cloud users' roles in cloud environments and their level of access to cloud resources in assessing cloud services and more precisely evaluating trust in cloud computing;

- Each category of cloud users (individual, organization, cloud service provider) possesses a separate model of measurable trust characteristics and sub-characteristics to facilitate

identifications of these characteristics to be evaluated in the processes of trust assessment in cloud computing based on cloud users' requirements with ED-BeCT (chapter 8).

- Results of chapter 8

The main result of this chapter is the details of ED-BeCT, its methodology and description of the phases to evaluate the provided cloud services. Based on the results obtained by applying this model, the trustworthy cloud service provider can be selected.

10.2 Research achievements

The main achievements, including contributions to the field can be summarised as follows:

- As explained in previous chapters and based on the literature, trust is an uncontrollable and intangible concept. Nonetheless by proposing ED-BeCT in this research project, we provide a method to measure this uncontrollable concept according to the applicable standards precisely;

- There is no doubt that cloud trust model characteristics are mostly based on customer requirements. Furthermore, to the best of our knowledge there is no special rule or agreed standard to select those characteristics and there is not much research in this area. In this research project by performing various analyses, we standardized trust characteristics with respect to criteria of the cloud computing standards. In addition, we combined system and software quality standards and cloud computing standards with the various perspectives of researchers in the literature and addressed several limitations in this regard which are already discussed in previous chapters;

- By categorizing cloud service users into three groups of individual, organization and the owner of the cloud services (provider), we attain the following achievements :

- We make the trust evaluation process in cloud environments feasible and easy;
- We propose a practical model (ED-BeCT);
- We decrease the complexity of the trust evaluation for each category of cloud users based on their level of access to the cloud resources.

In addition to these achievements, there are significant contributions that are explained in the next chapter.

10.3 The potential benefits of this research project for the industry

The proposed model in this research project creates a novel research avenue to assess the trustworthiness of the cloud service provider and will be a good reference for selection of the main criteria in this regard. In addition, it will help the International standard organizations to provide a comprehensive standard in trust evaluation in cloud computing technology.

Compared to previous methods for assessing trustworthiness of the cloud service providers, this research project introduced a complete set of criteria along with their measures to be considered as a standard reference, as all the characteristics are defined based on the ISO/IEC standards. This fact can have a positive impact on the cloud users and increase their level of confidence in trust assessment and provide an accurate method of evaluation.

Moreover, the results of this research have remarkable implications for the engineering community in the domain of cloud computing. The proposed model (ED-BeCT) by having simple mathematical procedure will offer the industry accuracy and precision in addition to facility for evaluating trust in cloud computing based on the standard characteristics that conform to their requirements. The use of ED-BeCT will provide the flexibility in assessing trust based on the various requirements and offer a method to deal with these requirements appropriately. This can be a good reason to promote motivation for adopting cloud computing technology which would be very profitable for cloud service providers and cloud service users.

Thus, the obtained results by using ED-BeCT can be beneficial in particular for the following groups :

- Cloud service users,
- Cloud service providers,
- Cloud auditors or professional organizations (standards or audit).

A summarized description of this research project and the main contributions are presented in the next chapter.

CONCLUSION

Summary of the research

In this thesis, we presented an enhanced dynamic behavioral cloud trust model (ED-BeCT) and described its various phases. The main goal of ED-BeCT is to assess the trustworthiness of the cloud service provider to support different types of cloud users in the process of the trustworthy cloud service provider selection. ED-BeCT is born from the ideas of : a- a motivation for cloud users to contribute to cloud service evaluations based on their perspective; b- supporting various users' requirements by focusing on the quality standards and cloud computing standards; c- presenting a simple and organized structure on how to distinguish trustable cloud service providers. It provides a complete set of generalized criteria for the three types of cloud service models along with the standard measures to address different aspects of trust in cloud computing. As explained, these criteria are the representatives of users' requirements, quality of services, different aspects of objective and subjective trust related to the cloud services and also different key intrinsic characteristics of cloud computing which are defined in ISO/IEC 17788 standard. Although, most of these characteristics are quality-based and are derived from software quality standards, to distinguish them from conventional software quality characteristics (given the cloud computing definition in NIST [2]), and also to emphasize on the fact that trust is not just quality of the cloud services, they are named as «trust characteristics » throughout this research project.

Basically, this thesis consists of four main parts. In the first part of this thesis, a literature review on trust characteristics was conducted to identify trust characteristics that were proposed in the domain-related research papers and those that are published in the broadly acceptable software and systems quality standards and cloud computing standards. In this research, in order to propose the key trust characteristics for cloud services, software quality and cloud-related standards trust characteristics and these commonly used and scientifically supported for trust evaluation of cloud service providers were identified. In the phase of matching both sets of characteristics, the common set of characteristics was identified. Finally,

the combination of the identified common set of trust characteristics with these from ISO/IEC 25010 and several cloud computing intrinsic characteristics proposed in published research led to the proposition of the minimal prototype model for cloud services trust.

In the second part of this thesis, we addressed some serious shortcomings of all the previously mentioned methods. First, it is important to consider the contribution of the related ISO/IEC standards when establishing measures for cloud trust characteristics. Second, there are various methods proposed by different researchers that are not comprehensive enough to evaluate trust characteristics in cloud environments, since they lack scientifically supported measures. Third, as the notion of trust also has uncertainty associated with it, by combining both subjective and objective evaluation of trust in cloud environments we can achieve a more credible outcome. Consequently, we analyzed cloud computing standards as well as system/software quality standards in order to extract measures that are potentially applicable in evaluation of the proposed trust characteristics in cloud computing.

In the third part of this thesis, we analyzed the extracted measures for the trust characteristics and based on the identified categorizations of cloud users (individual, organization, cloud service provider) in cloud environments, we specified trust characteristics to be measurable for the users categorized in these three categories for the evaluation purposes. Finally, we proposed a comprehensive behavioral cloud trust model and by presenting an example, we demonstrated the phases of this model to evaluate the provided cloud services.

The results of this evaluation, can be used for the following intentions:

- Supporting cloud users to select (a) qualified cloud service provider(s) : cloud users recognize the severity of the required quality characteristics related to the required cloud services. Based on this recognition, cloud users can have cloud service selection based on his/her interests and requirements among different proposed cloud services more confidently and decide for using or rejection of the offered cloud services;

- Supporting cloud auditors or professional organizations (standards or audit) or generally third parties : the obtained results by applying ED-BeCT can be helpful for the third parties that beside their cloud assessments, to consider the results of assessments from users' perspective in the processes of making a decision to grant (or not) the certification to the evaluated cloud services of a cloud service provider;
- Supporting cloud service providers in the evaluation purposes : cloud service providers by applying ED-BeCT can identify deficiencies and weaknesses in their proposed cloud services in order to proceed to improve them.

The major contributions

This thesis documents several key contributions made to the fields of trust evaluation in cloud computing:

- The trust characteristics in this research solution conform to the ISO/IEC standards and are as a result of rigorous analysis of system and software quality standards and cloud computing standards along with the literature. Hence, we identified some other components (apart from quality of service) that can directly impact on the trustworthiness of the cloud service providers such as auditability. We argue that QoS (Quality of Service) characteristics and these identified components can be used complementary to each other. Thus, we combined them for evaluating the trustworthiness of the cloud service provider. The identified set of characteristics can be considered as complete as possible trust characteristics reference in the time of conducting this research. Moreover, these characteristics can cover the other important criteria of the system such as resiliency that is related to fault tolerance and recoverability. However, most studies in the field of evaluating trust in cloud computing have only focused on the limited number of (QoS) characteristics as trust characteristics;
- To date, the existing literature on trust characteristics and their measures are mostly based on the authors' perspectives by their analysis of the literature. But this approach may not be comprehensive enough in order to evaluate trust in cloud environments, since there may be

some inconsistencies and discrepancies while proposing trust characteristics and their measures. Moreover, the measures we found in the literature for assessing the proposed trust characteristics do not conform to the system and software quality standards, nor are they easily applied to the context of cloud computing. Since there is no rule to select cloud trust characteristics measures, we chose instead to refer to existing standards and identify these measures. We chose these standards as they are agreed-upon reference, available, and have scientific and academic support. As part of this research, we identified the measures of trust characteristics based on analyzing standard characteristics of cloud computing and IT services, that most of them extracted from system and software quality standards and cloud computing standards. In addition, for the characteristics that there is no defined measures in the related standards, according to the methods explained in ISO/IEC 25021 standard, we develop proper measures based on their definition in ISO/IEC standards. So far, this is the first time that a cloud trust model can evaluate the standard trust characteristics based on evaluating their standard measures;

- The proposed trust characteristics along with their measures are generalized to be useful to evaluate the three types of cloud service models. So far, it is the first time that a cloud trust model is generalized for SaaS, PaaS and IaaS;
- In this research project, we also propose the measures for subjective trust. So far, to the best of our knowledge, it would be the first time that a cloud trust model evaluates the subjective trust through related measures as well as objective trust;
- In the proposed model, FUCOM which is a novel method and accurate technique with the maximum consistency and minimum deviation from full consistency to calculate the weights of the characteristics, is applied;
- The proposed model is able to compare the real behavior of the cloud service providers with their proposed SLAs and based on these results, the trustable cloud service providers will be filtered. Regarding the proposed measures, the level of trustworthiness of the filtered cloud service providers, can be measured as accurately as possible;
- This research, once officially published will be used by ISO/IEC JTC1 SC7 WG06 as the supporting material for the ongoing work on the applicability of ISO/IEC 25000 SQuaRE series to Cloud Computing technology. In addition, it will be used as the supporting material

in the works of ISO/IEC JTC1 SC7 WG06 Study Group on Software Quality Engineering domain within SQuRE chaired by Prof Witold Suryn, École de technologie superieure.

Several contributions documented in this research project have been published and submitted at a conference and journals. They are listed as follows :

Conference paper

Published

Moazzezi Eftekhar, W. Suryn, et al. (2018). "Towards the Development of a Widely Accepted Cloud Trust Model" was published in SQM Conference on Quality Management, 2018: p. 73.

Journal papers

Published

Eftekhar, S.M. and W. Suryn (2019). "A Proposition of Modifications and Extensions of Cloud Computing Standards for Trust Characteristics Measures" was published in Computer and Information Science, 2019. 12(3).

Limitations and challenges

ED-BeCT has been developed to assess the quality of the cloud services and support individual and organization to select a trustworthy cloud service provider. It is easy to apply but requires time to evaluate and monitor the elements of the proposed measures one by one based on the level of access to the cloud resources in cloud environment.

Moreover, this research is limited in several ways:

- ED-BeCT has been evaluated partially and it requires to be evaluated in real cases and in industrial context;

- There is a need for monitoring tools to support the evaluation processes in terms of assessing the measures;
- ED-BeCT does not define substitutes for unmeasurable measures;
- To be broadly acceptable, ED-BeCT requires to be evaluated by the experts to remove the weaknesses. Generally, this evaluation should be performed based on two steps : first, integration of ED-BeCT into the business setting of the organization in which ED-BeCT needs to be applied by identifying the business goals for which cloud services are required. Second, applying different phases of ED-BeCT that are mainly related to measuring required trust characteristics based on their identified measures for the executed cloud services to observe possible limitations. Moreover, in this part various sampling and testing should be taken into consideration to certify the applicability of the security-based measures in particular hacking interference. This part should be performed by the white hat hackers to test various scenarios.

Considerably more work will need to be done to address these limitations.

Future research

We hope that our research will serve as a base for future studies in the following areas:

- In some situations, the final selected cloud service provider would not be able to address all the users' requirements. Therefore, it needs to collaborate with the other cloud service providers in order to meet the users' demands. However, in most of the time cloud users are not aware of this fact and the users' data may geographically be distributed in various data centers. This fact can directly impact on the level of trust in cloud computing, as the evaluation processes are explicitly related to the cloud service provider with who cloud user has the contract. Figure 10-1 is related to the discussed example in chapter 9 and represents this situation clearly. As depicted, this is a hybrid cloud scenario in which a hybrid cloud is delivered to the individuals (the PACS vendor enterprise and its potential customers that are hospitals and medical centers) by a federated cloud service provider that its own services

are combined with the other cloud services provided by different cloud service providers [272]. But this fact is not transparent for the PACS vendor. This is very much the key component in future attempts to overcome this issue;

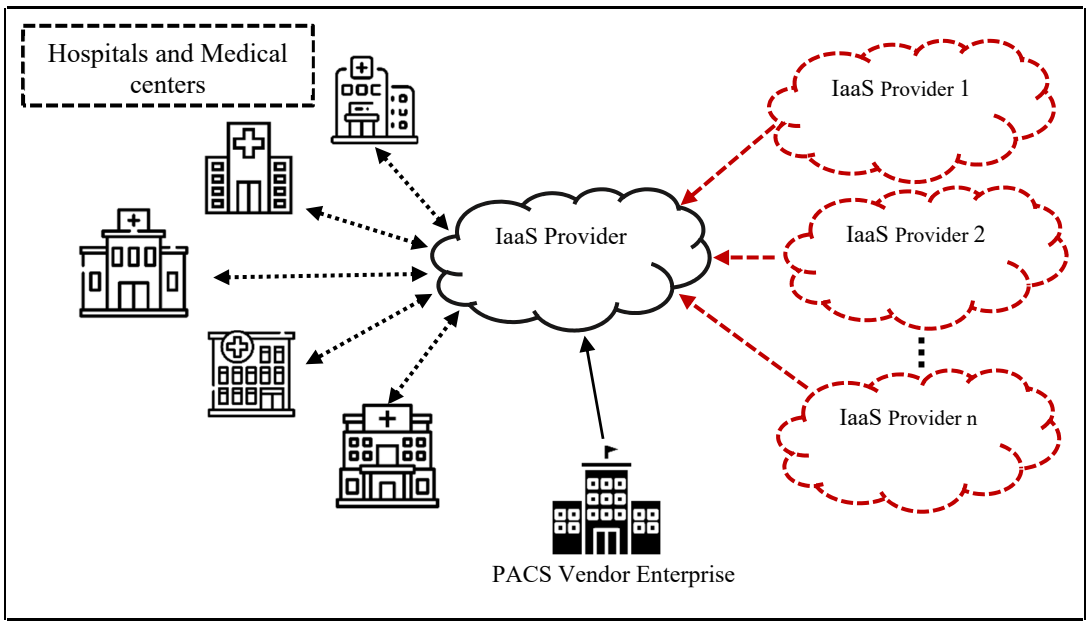

Figure 11-1. A sample schematic of cloud service usage related to the discussed example in chapter 9. In this case, the PACS vendor enterprise is not aware of other IaaS providers represented in dotted-red line. Hospitals and medical centers are the potential customers of the PACS vendor enterprise.

- On a wider level, research is also needed to determine a proper method to evaluate unmeasurable measures, in particular buffer over flow that is one of the measures of integrity and the measures for resource utilization in cloud environments;

- Future studies on the current topic are suggested in order to create additional characteristics and/or measures by using ISO/IEC 25000 series regarding the cloud computing criteria defined in the cloud technology standards to increase the trust assessment accuracy.

REFERENCES

1. ISO/IEC, *25010 (2011) Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models.* International Organization for Standardization, Geneva, Switzerland, 2011.

2. Mell, P. and Grance,T , *The NIST definition of cloud computing.* 2011.

3. Almarabeh, T., Majdalawi.Y, and Mohammad.H, *Cloud computing of e-government.* 2016.

4. Furuncu, E. and Sogukpinar.I , *Scalable risk assessment method for cloud computing using game theory (CCRAM).* Computer Standards & Interfaces, 2015. **38**: p. 44-50.

5. Herbst, N.R., Kounev.R , and Reussner,V. *Elasticity in cloud computing: What it is, and what it is not*. in *Proceedings of the 10th International Conference on Autonomic Computing ({ICAC} 13)*. 2013.

6. Lehrig, S., Eikerling. H, and Becker.S. *Scalability, elasticity, and efficiency in cloud computing: A systematic literature review of definitions and metrics*. in *Proceedings of the 11th International ACM SIGSOFT Conference on Quality of Software Architectures*. 2015. ACM.

7. Rekik, M., Boukadi.K, and Ben-Abdallah.H. *A comprehensive framework for business process outsourcing to the cloud*. in *2016 IEEE International Conference on Services Computing (SCC)*. 2016. IEEE.

8. Abbadi, I.M. and Martin.A, *Trust in the Cloud.* information security technical report, 2011. **16**(3-4): p. 108-114.

9. Habib, S.M., et al., *Trust as a facilitator in cloud computing: a survey.* Journal of Cloud Computing: Advances, Systems and Applications, 2012. **1**(1): p. 19.

10. Sarwar, A. and Khan.M.N, *A review of trust aspects in cloud computing security.* International Journal of Cloud Computing and Services Science, 2013. **2**(2): p. 116.

11. Manuel, P., *A trust model of cloud computing based on quality of service.* Annals of Operations Research, 2015. **233**: p. 281-92.

12. Huang, J. and Nicol, D.M. *Trust mechanisms for cloud computing.* Journal of Cloud Computing: Advances, Systems and Applications, 2013. **2**(1): p. 9.

13. Moazzezi Eftekhar, S. and Suryn, W. , *A Proposition of Modifications and Extensions of Cloud Computing Standards for Trust Characteristics Measures.* Computer and Information Science, 2019. **12**.

14. MoazzeziEftekhar, S. and Suryn,W. *A Proposition of Modifications and Extensions of Cloud Computing Standards for Trust Characteristics Measures.* Computer and Information Science, 2019. **12**(3).

15. Singh, S. and Sidhu,J., *Compliance-based multi-dimensional trust evaluation system for determining trustworthiness of cloud service providers.* Future Generation Computer Systems, 2017. **67**: p. 109-132.

16. Fan, W.-J., et al., *A multi-dimensional trust-aware cloud service selection mechanism based on evidential reasoning approach.* International Journal of Automation and Computing, 2015. **12**(2): p. 208-219.

17. Sztompka, P., *Trust: A sociological theory*. 1999: Cambridge University Press.

18. Özer, Ö. and Zheng, Y., *Establishing trust and trustworthiness for supply chain information sharing*, in *Handbook of Information Exchange in Supply Chain Management*. 2017, Springer. p. 287-312.

19. Suryn, W., *Software quality engineering: a practitioner's approach*. 2013: John Wiley & Sons.

20. Moazzezi Eftekhar, S., et al., *Towards the Development of a Widely Accepted Cloud Trust Model.* SQM XXVI, 2018: p. 73.

21. Chahal, R.K. and Singh, S , *Fuzzy rule-based expert system for determining trustworthiness of cloud service providers.* International Journal of Fuzzy Systems, 2017. **19**(2): p. 338-354.

22. Li, Z., et al., *Evaluating the credibility of cloud services.* Computers & Electrical Engineering, 2017. **58**: p. 161-175.

23. Rajendran, V.V. and Swamynathan, S., *Hybrid model for dynamic evaluation of trust in cloud services.* Wireless Networks, 2016. **22**(6): p. 1807-1818.

24. ISO/IEC, *17788 (2014)Information technology — Cloud computing — Overview and vocabulary.* International Organization for Standardization, Geneva, Switzerland, 2014.

25. Gonzales, D., et al., *Cloud-trust—A security assessment model for infrastructure as a service (IaaS) clouds.* IEEE Transactions on Cloud Computing, 2017. **5**(3): p. 523-536.

26. Bo, T. and Sandhu, R. *Cross-tenant trust models in cloud computing*. in *2013 IEEE 14th International Conference on Information Reuse & Integration (IRI), 14-16 Aug. 2013*. 2013. Piscataway, NJ, USA: IEEE.

27. Farcasescu, M.R. *Trust model engines in cloud computing*. in *2012 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2012), 26-29 Sept. 2012*. 2012. Los Alamitos, CA, USA: IEEE Computer Society.

28. Fernandes, D.A.B., et al., *Security issues in cloud environments: a survey.* International Journal of Information Security, 2014. **13**(2): p. 113-70.

29. Karthiga, R., et al. *Supporting reputation based trust management enhancing security layer for cloud service models*. in *14th International Conference on Science, Engineering and Technology (ICSET-2017), 2-3 May 2017*. 2017. UK: IOP Publishing.

30. Li, W. and Ping, L. *Trust model to enhance security and interoperability of cloud environment*. in *1st International Conference on Cloud Computing, CloudCom 2009, December 1, 2009 - December 4, 2009*. 2009. Beijing, China: Springer Verlag.

31. Rizvi, S., et al. *A centralized trust model approach for cloud computing*. in *Wireless and Optical Communication Conference (WOCC), 2014 23rd*. 2014. IEEE.

32. Shaikh, R.A.R. and Sasikumar, M. *Trust model for a cloud computing application and service*. in *2012 3rd IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2012, December 18, 2012 - December 20, 2012*. 2012. Coimbatore, Tamilnadu, India: IEEE Computer Society.

33. Wang, W., et al. *The design of a trust and role based access control model in cloud computing*. in *Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on*. 2011. IEEE.

34. Wu, X., et al., *A trust evaluation model for cloud computing.* Procedia Computer Science, 2013. **17**: p. 1170-1177.

35. Zhao, G., et al. *Deployment models: Towards eliminating security concerns from cloud computing*. in *High Performance Computing and Simulation (HPCS), 2010 International Conference on*. 2010. IEEE.

36. Habib, S.M., Varadharajan, V., and Mühlhäuser, M. *A trust-aware framework for evaluating security controls of service providers in cloud marketplaces*. in *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*. 2013. IEEE.

37. Shaikh, R. and Sasikumar, M., *Trust model for measuring security strength of cloud computing service.* Procedia Computer Science, 2015. **45**: p. 380-389.

38. Ghosh, N., Ghosh, S.K., and Das, S.K., *SelCSP: A framework to facilitate selection of cloud service providers.* IEEE transactions on cloud computing, 2015. **3**(1): p. 66-79.

39. Divakarla, U. and Sekaran, K.C. *Trust models in cloud: A survey on pros and cons.* Lecture Notes in Electrical Engineering, 2015. **312**: p. 335-341.

40. Abdallah, E.G., et al. *TRUST-CAP: A Trust Model for Cloud-Based Applications*. in *Computer Software and Applications Conference (COMPSAC), 2017 IEEE 41st Annual*. 2017. IEEE.

41. Singh, A. and Chatterjee, K. *A multi-dimensional trust and reputation calculation model for cloud computing environments*. in *2017 ISEA Asia Security and Privacy (ISEASP), 29 Jan.-1 Feb. 2017*. 2017. Piscataway, NJ, USA: IEEE.

42. Li, X. and Du, J., *Adaptive and attribute-based trust model for service-level agreement guarantee in cloud computing.* IET Information Security, 2013. **7**(1): p. 39-50.

43. Chu, R., Lai, I.K.W. and Lai, D.C.F. *Trust factors influencing the adoption of cloud-based interorganizational systems: a conceptual model*. in *2013 International Conference on Engineering, Management Science and Innovation (ICEMSI), 28-30 June 2013*. 2013. Piscataway, NJ, USA: IEEE.

44. Garg, S.K., Versteeg, S. and Buyya, R. *A framework for ranking of cloud computing services.* Future Generation Computer Systems, 2013. **29**(4): p. 1012-23.

45. Kai, Y., Ying, C. and Fei, T. *A trust evaluation model towards cloud manufacturing.* International Journal of Advanced Manufacturing Technology, 2016. **84**(1-4): p. 133-46.

46. Habib, S.M., Ries, S. and Muhlhauser, M. *Towards a trust management system for cloud computing*. in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2011 IEEE 10th International Conference on*. 2011. IEEE.

47. Selvaraj, A. and Sundararajan, S. *Evidence-Based Trust Evaluation System for Cloud Services Using Fuzzy Logic.* International Journal of Fuzzy Systems, 2017. **19**(2): p. 329-337.

48. Marudhadevi, D., Dhatchayani, V.N. and Sriram, V.S. *A trust evaluation model for cloud computing using service level agreement.* The Computer Journal, 2014. **58**(10): p. 2225-2232.

49. Hajizadeh, R. and Jafari Navimipour N., *A method for trust evaluation in the cloud environments using a behavior graph and services grouping.* Kybernetes, 2017. **46**(7): p. 1245-1261.

50. Ritu and Jain, S. *A trust model in cloud computing based on fuzzy logic*. in *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 20-21 May 2016*. 2016. Piscataway, NJ, USA: IEEE.

51. Lin, G., et al., *ACO-BTM: a behavior trust model in cloud computing environment.* International Journal of Computational Intelligence Systems, 2014. **7**(4): p. 785-795.

52. Zhang, P., Kong, Y. and Zhou, M. *A domain partition-based trust model for unreliable clouds.* IEEE Transactions on Information Forensics and Security, 2018. **13**(9): p. 2167-2178.

53. Noor, T.H., et al. *Cloud armor: a platform for credibility-based trust management of cloud services*. in *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. 2013. ACM.

54. Kanwal, A., et al., *Taxonomy for trust models in cloud computing.* Computer Journal, 2015. **58**(4): p. 601-626.

55. Masaharu, S., *Personal data in the cloud, the importance of trust.* Fuijitsu Research Institute, 2010.

56. Khan, K.M. and Malluhi, Q. *Establishing trust in cloud computing.* IT professional, 2010. **12**(5): p. 20-27.

57. Pearson, S. and Benameur, A. *Privacy, security and trust issues arising from cloud computing*. in *2010 IEEE Second International Conference on Cloud Computing Technology and Science*. 2010. IEEE.

58. Pawar, P.S., et al. *Trust model for optimized cloud services*. in *IFIP international conference on trust management*. 2012. Springer.

59. Subashini, S. and Kavitha, V. *A survey on security issues in service delivery models of cloud computing.* Journal of network and computer applications, 2011. **34**(1): p. 1-11.

60. Alhanahnah, M., et al., *Context-aware multifaceted trust framework for evaluating trustworthiness of cloud providers.* Future Generation Computer Systems, 2018. **79**: p. 488-499.

61. Lynn, T., et al., *A delphi approach to the development of a cloud trust label.* Journal of Computer Information Systems, vol, 2015.

62. Emeakaroha, V.C., et al., *A Trust Label System for Communicating Trust in Cloud Services.* IEEE Transactions on Services Computing, 2017. **10**(5): p. 689-700.

63. Söllner, M., Hoffmann, A. and Leimeister, J.M. *Why different trust relationships matter for information systems users.* European Journal of Information Systems, 2016. **25**(3): p. 274-287.

64. Söllner, M., Pavlou, P. and Leimeister, J. *Understanding trust in IT Artifacts–a new conceptual approach.* 2013.

65. Moyano, F., Fernandez-Gago, C. and Lopez, J. *A conceptual framework for trust models*. in *International Conference on Trust, Privacy and Security in Digital Business*. 2012. Springer.

66. Thampi, S.M., Atrey, P.K. and Bhargava, B. *Managing trust in cyberspace*. 2013: Chapman and Hall/CRC.

67. Nagarajan, A., Varadharajan, V. and Tarr, N. *Trust enhanced distributed authorisation for web services.* Journal of Computer and System Sciences, 2014. **80**(5): p. 916-934.

68. Jøsang, A., Ismail, R. and Boyd, C. *A survey of trust and reputation systems for online service provision.* Decision support systems, 2007. **43**(2): p. 618-644.

69. Rimal, B.P., Choi, E. and Lumb. I. *A taxonomy and survey of cloud computing systems*. in *2009 Fifth International Joint Conference on INC, IMS and IDC*. 2009. Ieee.

70. Hwang, K., Kulkareni, S. and Hu, Y. *Cloud security with virtualized defense and reputation-based trust mangement*. in *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*. 2009. IEEE.

71. Liu, Y.-C., et al., *A method for trust management in cloud computing: Data coloring by cloud watermarking.* International journal of automation and computing, 2011. **8**(3): p. 280.

72. Sun, X., Chang, G. and Li, F. *A trust management model to enhance security of cloud computing environments*. in *2011 Second International Conference on Networking and Distributed Computing*. 2011. IEEE.

73. Tang, M., et al., *Towards a trust evaluation middleware for cloud service selection.* Future Generation Computer Systems, 2017. **74**: p. 302-312.

74. Chiregi, M. and Navimipour, N.J. *A comprehensive study of the trust evaluation mechanisms in the cloud computing.* Journal of Service Science Research, 2017. **9**(1): p. 1-30.

75. Wang, Y., et al. *A novel dynamic cloud service trust evaluation model in cloud computing*. in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. 2018. IEEE.

76. Agheli, N., Hosseini, B. and Shojaee, A. *A trust evaluation model for selecting service provider in cloud environment*. in *2014 4th International Conference on Computer and Knowledge Engineering (ICCKE)*. 2014. IEEE.

77. Matos, R., Marques, C. and Sargento, S. *Context-aware control of user-centric virtual networks: Centralized vs distributed approaches.* Computer Networks, 2014. **74**: p. 4-21.

78. Navimipour, N.J., et al., *Resource discovery mechanisms in grid systems: A survey.* Journal of Network and Computer Applications, 2014. **41**: p. 389-410.

79. Jabbar, S., et al., *Trust model at service layer of cloud computing for educational institutes.* The Journal of Supercomputing, 2016. **72**(1): p. 58-83.

80. Emeakaroha, V.C., et al., *A trust label system for communicating trust in cloud services.* IEEE Transactions on Services Computing, 2016. **10**(5): p. 689-700.

81. Zou, D., et al., *Design and implementation of a trusted monitoring framework for cloud platforms.* Future Generation Computer Systems, 2013. **29**(8): p. 2092-2102.

82. Ko, R.K., et al. *TrustCloud: A framework for accountability and trust in cloud computing*. in *2011 IEEE World Congress on Services*. 2011. IEEE.

83. Marsh, S.P., *Formalising trust as a computational concept.* 1994.

84. Manuel, P., *A trust model of cloud computing based on Quality of Service.* Annals of Operations Research, 2015. **233**(1): p. 281-292.

85. Fan, W. and Perros, H. *A novel trust management framework for multi-cloud environments based on trust service providers.* Knowledge-Based Systems, 2014. **70**: p. 392-406.

86. Deshpande, S. and Ingle, R. *Trust assessment in cloud environment: Taxonomy and analysis*. in *2016 International Conference on Computing, Analytics and Security Trends (CAST), 19-21 Dec. 2016*. 2016. Piscataway, NJ, USA: IEEE.

87. Rawashdeh, E.F., Abuqaddom, I.I. and Hudaib, A.A. *Trust models for services in cloud environment: A survey*. in *2018 9th International Conference on Information and Communication Systems (ICICS)*. 2018. IEEE.

88. Habib, S.M., Ries, S. and Muhlhauser, M. *Towards a trust management system for cloud computing*. in *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*. 2011. IEEE.

89. Alhamad, M., Dillon, T. and Chang, E. *Sla-based trust model for cloud computing*. in *2010 13th international conference on network-based information systems*. 2010. IEEE.

90. Mendel, J.M., *Fuzzy logic systems for engineering: a tutorial.* Proceedings of the IEEE, 1995. **83**(3): p. 345-377.

91. Noor, T.H., et al., *Trust management of services in cloud environments: Obstacles and solutions.* ACM Computing Surveys (CSUR), 2013. **46**(1): p. 12.

92. Khan, M.S., Warsi, M. and Islam, S. *Trust Management Issues in Cloud Computing Ecosystems.* Available at SSRN 3358749, 2019.

93. El Kassabi, H.T. and Serhani, M.A. *De-centralized reputation-based trust model to discriminate between cloud providers capable of processing big data*. in *2017 IEEE International Congress on Big Data (BigData Congress)*. 2017. IEEE.

94. Resnick, P., et al., *Reputation systems.* Communications of the ACM, 2000. **43**(12): p. 45-48.

95. Kanwal, A., et al. *Assessment Criteria for Trust Models in Cloud Computing*. in *2013 IEEE International Conference on Green Computing and Communications (GreenCom) and IEEE Internet of Things (iThings) and IEEE Cyber, Physical and Social Computing (CPSCom), 20-23 Aug. 2013*. 2013. Los Alamitos, CA, USA: IEEE Computer Society.

96. Krautheim, F.J. *Private Virtual Infrastructure for Cloud Computing*. in *HotCloud*. 2009.

97. Sato, H., Kanai, A. and Tanimoto, S. *A cloud trust model in a security aware cloud*. in *2010 10th IEEE/IPSJ International Symposium on Applications and the Internet*. 2010. IEEE.

98. Ahmed, M. and Xiang, Y. *Trust ticket deployment: a notion of a data owner's trust in cloud computing*. in *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*. 2011. IEEE.

99. Bezzi, M., Kaluvuri, S.P. and Sabetta, A. *Ensuring trust in service consumption through security certification*. in *Proceedings of the International Workshop on Quality Assurance for Service-Based Applications*. 2011. ACM.

100. Krautheim, F.J., Phatak, D.S. and Sherman, A.T. *Introducing the trusted virtual environment module: a new mechanism for rooting trust in cloud computing*. in *International Conference on Trust and Trustworthy Computing*. 2010. Springer.

101. Firdhous, M., Ghazali, O. and Hassan, S. *A trust computing mechanism for cloud computing*. in *Proceedings of ITU Kaleidoscope 2011: The Fully Networked Human?-Innovations for Future Networks and Services (K-2011)*. 2011. IEEE.

102. Noor, T.H. and Sheng, Q.Z. *Trust as a service: A framework for trust management in cloud environments*. in *International Conference on Web Information Systems Engineering*. 2011. Springer.

103. Mrabet, M., ben Saied, Y. and Saidane, L.A. *A new trust evaluation approach for cloud computing environments*. in *2016 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*. 2016. IEEE.

104. Li, W. and Ping, L. *Trust model to enhance security and interoperability of cloud environment*. in *IEEE International Conference on Cloud Computing*. 2009. Springer.

105. Yang, Z., et al. *A collaborative trust model of firewall-through based on Cloud Computing*. in *The 2010 14th International Conference on Computer Supported Cooperative Work in Design*. 2010. IEEE.

106. Zhou, Z.-x., Xu, H. and Wang, S.-p. *A novel weighted trust model based on cloud.* Advances in Information Sciences and Service Sciences, 2011. **3**(3): p. 115-124.

107. Li, W., et al., *Research on trust management strategies in cloud computing environment.* Journal of Computational Information Systems, 2012. **8**(4): p. 1757-1763.

108. Hajizadeh, R. and Jafari Navimipour, N. *A method for trust evaluation in the cloud environments using a behavior graph and services grouping.* Kybernetes, 2017(just-accepted): p. 00-00.

109. Raman, M.G., et al., *A hypergraph and arithmetic residue-based probabilistic neural network for classification in intrusion detection systems.* Neural Networks, 2017. **92**: p. 89-97.

110. Somu, N., Kirthivasan, K. and Sriram, V.S. *A rough set-based hypergraph trust measure parameter selection technique for cloud service selection.* The Journal of Supercomputing, 2017. **73**(10): p. 4535-4559.

111. Somu, N., Kirthivasan, K. and VS, S.S. *A computational model for ranking cloud service providers using hypergraph based techniques.* Future Generation Computer Systems, 2017. **68**: p. 14-30.

112. Bretto, A., *Hypergraph theory.* An introduction. Mathematical Engineering. Cham: Springer, 2013.

113. http://www.nickgillian.com/wiki/pmwiki.php/GRT/MinDist.

114. https://www.statisticshowto.datasciencecentral.com/.

115. Pawlak, Z., et al., *Rough sets.* Communications of the ACM, 1995. **38**(11): p. 88-95.

116. Pawlak, Z. and Skowron, A. *Rudiments of rough sets.* Information sciences, 2007. **177**(1): p. 3-27.

117. Rissino, S. and Lambert-Torres, G. *Rough set theory—fundamental concepts, principals, data extraction, and applications*, in *Data mining and knowledge discovery in real life applications*. 2009, InTech.

118. Nabwey, H.A., *A probabilistic rough set approach to rule discovery.* International Journal of Advanced Science and Technology, 2011. **30**: p. 25-34.

119. León, T., et al., *Using Induced Ordered Weighted Averaging (IOWA) Operators for Aggregation in Cross-Efficiency Evaluations.* International Journal of Intelligent Systems, 2014. **29**(12): p. 1100-1116.

120. Xian, S., et al., *Fuzzy linguistic induced OWA Minkowski distance operator and its application in group decision making.* Pattern Analysis and Applications, 2016. **19**(2): p. 325-335.

121. Merigó, J.M. and Gil-Lafuente, A.M. *The induced generalized OWA operator.* Information Sciences, 2009. **179**(6): p. 729-741.

122. Torra, V. and Narukawa, Y. *Modeling decisions: information fusion and aggregation operators*. 2007: Springer Science & Business Media.

123. Marin, L., et al., *Induced unbalanced linguistic ordered weighted average and its application in multiperson decision making.* The Scientific World Journal, 2014. **2014**.

124. Yager, R.R., *On ordered weighted averaging aggregation operators in multicriteria decisionmaking.* IEEE Transactions on systems, Man, and Cybernetics, 1988. **18**(1): p. 183-190.

125. Zimmermann, H.J., *Fuzzy set theory.* Wiley Interdisciplinary Reviews: Computational Statistics, 2010. **2**(3): p. 317-332.

126. Nagarajan, R., Selvamuthukumaran, S. and Thirunavukarasu, R. *A fuzzy logic based trust evaluation model for the selection of cloud services*. in *2017 International Conference on Computer Communication and Informatics (ICCCI)*. 2017. IEEE.

127. Chahal, R.K. and Singh, S. *Trust Calculation Using Fuzzy Logic in Cloud Computing*, in *Handbook of Research on Security Considerations in Cloud Computing*. 2015, IGI Global. p. 127-172.

128. Nafi, K.W., Hossain, A. and Hashem, M. *An advanced certain trust model using fuzzy logic and probabilistic logic theory.* arXiv preprint arXiv:1303.0459, 2013.

129. Alabool, H.M. and Mahmood, A.K. *Trust-based service selection in public cloud computing using fuzzy modified VIKOR method.* Australian Journal of Basic and Applied Sciences, 2013. **7**(9): p. 211-220.

130. Qu, C. and Buyya, R. *A cloud trust evaluation system using hierarchical fuzzy inference system for service selection*. in *Advanced information networking and applications (aina), 2014 ieee 28th international conference on*. 2014. IEEE.

131. Godil, S.S., et al., *Fuzzy logic: A "simple" solution for complexities in neurosciences?* Surgical neurology international, 2011. **2**.

132. https://brilliant.org/wiki/markov-chains/.

133. Chandrasekar, A., et al. *QoS monitoring and dynamic trust establishment in the cloud*. in *International Conference on Grid and Pervasive Computing*. 2012. Springer.

134. Skalny, P. and Krajc, B. *Discrete-Time Markov Chains in Reliability Analysis-Case Study*. in *International Joint Conference CISIS'12-ICEUTE´ 12-SOCO´ 12 Special Sessions*. 2013. Springer.

135. https://www.statisticssolutions.com/pearsons-correlation-coefficient/.

136. Tan, W., et al., *A trust service-oriented scheduling model for workflow applications in cloud computing.* IEEE Systems Journal, 2014. **8**(3): p. 868-878.

137. Pamučar, D., Stević, Ž. and Sremac, S. *A new model for determining weight coefficients of criteria in mcdm models: Full consistency method (fucom).* Symmetry, 2018. **10**(9): p. 393.

138. Sourani, A. and Sohail, M. *The Delphi method: Review and use in construction management research.* International Journal of Construction Education and Research, 2015. **11**(1): p. 54-76.

139. Iqbal, S. and Pipon-Young, L. *Methods-The Delphi method--A guide from Susanne Iqbal and Laura Pipon-Young.* Psychologist, 2009. **22**(7): p. 598.

140. Lynn, T., et al., *Development of a cloud trust label: a Delphi approach.* Journal of Computer Information Systems, 2016. **56**(3): p. 185-193.

141. Teknomo, K., *Analytic hierarchy process (AHP) tutorial.* Revoledu. com, 2006: p. 1-20.

142. Saaty, T.L., *How to make a decision: the analytic hierarchy process.* European journal of operational research, 1990. **48**(1): p. 9-26.

143. Sidhu, J. and Singh, S. *Improved topsis method based trust evaluation framework for determining trustworthiness of cloud service providers.* Journal of Grid Computing, 2017. **15**(1): p. 81-105.

144. Macharis, C., et al., *PROMETHEE and AHP: The design of operational synergies in multicriteria analysis.: Strengthening PROMETHEE with ideas of AHP.* European Journal of Operational Research, 2004. **153**(2): p. 307-317.

145. Siadat, S., Rahmani, A.M. and Navid, H. *Identifying fake feedback in cloud trust management systems using feedback evaluation component and Bayesian game model.* The Journal of Supercomputing, 2017. **73**(6): p. 2682-2704.

146. Dabade, T.D. *Information technology infrastructure library (ITIL)*. in *Proceedings of the 4th National Conference*. 2012.

147. Cardoso, A., Moreira, F. and Escudero, D.F. *Information Technology Infrastructure Library and the migration to cloud computing.* Universal Access in the Information Society, 2018. **17**(3): p. 503-515.

148. ISO/IEC, *25011 (2017) - Information technology — Systems and software quality requirements and evaluation (SQuaRE) — Service quality models.* International Organization for Standardization, Geneva, Switzerland, 2017.

149. ISO/IEC, *25012 (2008)Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE) - Data Quality Model.* International Organization for Standardization, Geneva, Switzerland, 2008.

150. ISO/IEC, *19086-1,Information technology — Cloud computing — Service level agreement (SLA) framework and technology — Part 1: Overview and concepts.* International Organization for Standardization, Geneva, Switzerland, 2016.

151. Noor, T.H., et al., *CloudArmor: Supporting reputation-based trust management for cloud services.* IEEE transactions on parallel and distributed systems, 2016. **27**(2): p. 367-380.

152. https://en.wikiversity.org/wiki/Software_metrics_and_measurement.

153. Fehling, C., et al., *Cloud computing patterns: fundamentals to design, build, and manage cloud applications*. 2014: Springer.

154. Ochei, L.C., Bass, J.M. and Petrovski, A. *Evaluating degrees of multitenancy isolation: A case study of cloud-hosted gsd tools*. in *Cloud and Autonomic Computing (ICCAC), 2015 International Conference on*. 2015. IEEE.

155. AlJahdali, H., et al. *Multi-tenancy in cloud computing*. in *Service Oriented System Engineering (SOSE), 2014 IEEE 8th International Symposium on*. 2014. IEEE.

156. Khan, M.F. and Ullah, M.A. *An approach towards customized multi-tenancy.* International Journal of Modern Education and Computer Science (IJMECS)[online], 2012. **4**(9): p. 39.

157. Bezemer, C.-P. and Zaidman, A. *Multi-tenant SaaS applications: maintenance dream or nightmare?* in *Proceedings of the Joint ERCIM Workshop on Software Evolution (EVOL) and International Workshop on Principles of Software Evolution (IWPSE)*. 2010. ACM.

158. Villalpando, L.E.B., et al., *A Three-Dimensional Performance Measurement Model for Cloud Computing.* Journal of Software Engineering and Applications, 2018. **11**(05): p. 235.

159. Ataş, G. and Gungor, V.C. *Performance evaluation of cloud computing platforms using statistical methods.* Computers & Electrical Engineering, 2014. **40**(5): p. 1636-1649.

160. ISO/IEC, *25023 (2016) Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality.* International Organization for Standardization, Geneva, Switzerland, 2016.

161. Halabi, T. and Bellaiche, M. *Towards quantification and evaluation of security of Cloud Service Providers.* Journal of Information Security and Applications, 2017. **33**: p. 55-65.

162. Abdel-Basset, M., Mohamed, M. and Chang, V. *NMCDA: A framework for evaluating cloud computing services.* Future Generation Computer Systems, 2018. **86**: p. 12-29.

163. Lehrig, S., et al., *CloudStore—towards scalability, elasticity, and efficiency benchmarking and analysis in Cloud computing.* Future Generation Computer Systems, 2018. **78**: p. 115-126.

164. Kuhlenkamp, J., Klems, M. and Röss, O. *Benchmarking scalability and elasticity of distributed database systems.* Proceedings of the VLDB Endowment, 2014. **7**(12): p. 1219-1230.

165. Coutinho, E.F., et al., *Physics and microeconomics-based metrics for evaluating cloud computing elasticity.* Journal of Network and Computer Applications, 2016. **63**: p. 159-172.

166. Sharma, Y., et al., *Reliability and energy efficiency in cloud computing systems: Survey and taxonomy.* Journal of Network and Computer Applications, 2016. **74**: p. 66-85.

167. Cui, H., et al., *Cloud service reliability modelling and optimal task scheduling.* IET Communications, 2017. **11**(2): p. 161-167.

168. Heckmann, I., Comes, T. and Nickel, S. *A critical review on supply chain risk–Definition, measure and modeling.* Omega, 2015. **52**: p. 119-132.

169. Ali, A., Warren, D. and Mathiassen, L. *Cloud-based business services innovation: A risk management model.* International Journal of Information Management, 2017. **37**(6): p. 639-649.

170. Venters, W. and Whitley, E.A. *A critical review of cloud computing: researching desires and realities.* Journal of Information Technology, 2012. **27**(3): p. 179-197.

171. Djemame, K., et al., *A risk assessment framework for cloud computing.* IEEE Transactions on Cloud Computing, 2016(1): p. 1-1.

172. Misra, K.B., *Handbook of performability engineering*. 2008: Springer Science & Business Media.

173. ISO/IEC, *25022 (2016) Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use.* International Organization for Standardization, Geneva, Switzerland, 2016.

174. Roy, S., Pattnaik, P.K. and Mall, R. *Quality assurance of academic websites using usability testing: an experimental study with AHP.* International Journal of System Assurance Engineering and Management, 2017. **8**(1): p. 1-11.

175. Stanton, B., Theofanos, M. and Joshi, K.P. *Framework for cloud usability*. in *International Conference on Human Aspects of Information Security, Privacy, and Trust*. 2015. Springer.

176. Pandey, S. and Daniel, A. *Fuzzy logic based cloud service trustworthiness model*. in *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*. 2016. IEEE.

177. Baharuddin, R., Singh, D. and Razali, R. *Usability dimensions for mobile applications—a review.* Res. J. Appl. Sci. Eng. Technol, 2013. **5**(6): p. 2225-2231.

178. Thomas, M.O., Onyimbo, B.A. and Logeswaran, R. *Usability evaluation criteria for internet of things.* International Journal of Information Technology and Computer Science, 2016. **8**(12): p. 10-18.

179. Lin, A. and Chen, N.-C. *Cloud computing as an innovation: Percepetion, attitude, and adoption.* International Journal of Information Management, 2012. **32**(6): p. 533-540.

180. Gangwar, H., Date, H. and Ramaswamy, R. *Understanding determinants of cloud computing adoption using an integrated TAM-TOE model.* Journal of Enterprise Information Management, 2015. **28**(1): p. 107-130.

181. Dastjerdi, A.V. and Buyya, R. *Compatibility-aware cloud service composition under fuzzy preferences of users.* IEEE Transactions on Cloud Computing, 2014. **2**(1): p. 1-13.

182. Fiandrino, C., et al., *Performance and energy efficiency metrics for communication systems of cloud computing data centers.* IEEE Transactions on Cloud Computing, 2017. **5**(4): p. 738-750.

183. Qiu, X., et al., *A hierarchical correlation model for evaluating reliability, performance, and power consumption of a cloud service.* IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2016. **46**(3): p. 401-412.

184. Rani, D. and Ranjan, R.K. *A comparative study of SaaS, PaaS and IaaS in cloud computing.* International Journal of Advanced Research in Computer Science and Software Engineering, 2014. **4**(6).

185. Sowmya, S., Deepika, P. and Naren, J. *Layers of cloud–iaas, paas, and saas: A survey.* International Journal of Computer Science and Information Technologies, 2014. **5**(3): p. 4477-4480.

186. Freet, D., et al. *Cloud forensics challenges from a service model standpoint: IaaS, PaaS and SaaS*. in *Proceedings of the 7th International Conference on Management of computational and collective intElligence in Digital EcoSystems*. 2015. ACM.

187. Jain, N. and Choudhary, S. *Overview of virtualization in cloud computing*. in *2016 Symposium on Colossal Data Analysis and Networking (CDAN)*. 2016. IEEE.

188. Sharma, S., et al., *Virtualization in Cloud Computing.* Int. J. Sci. Res. Sci. Eng. Technol, 2016. **2**: p. 181-186.

189. Blair, G., Schmidt, D. and Taconet, C. *Middleware for internet distribution in the context of cloud computing and the internet of things*. 2016, Springer.

190. Bhardwaj, S., Jain, L. and Jain, S. *Cloud computing: A study of infrastructure as a service (IAAS).* International Journal of engineering and information Technology, 2010. **2**(1): p. 60-63.

191. Adhikari, M. and Amgoth, T. *Heuristic-based load-balancing algorithm for IaaS cloud.* Future Generation Computer Systems, 2018. **81**: p. 156-165.

192. Rodriguez, M.A. and Buyya, R. *A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments.* Concurrency and Computation: Practice and Experience, 2017. **29**(8): p. e4041.

193. Manvi, S.S. and Shyam, G.K. *Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey.* Journal of network and computer applications, 2014. **41**: p. 424-440.

194. Dillon, T., Wu, C. and Chang, E. *Cloud computing: issues and challenges*. in *2010 24th IEEE international conference on advanced information networking and applications*. 2010. Ieee.

195. Pahl, C., *Containerization and the paas cloud.* IEEE Cloud Computing, 2015. **2**(3): p. 24-31.

196. Jadeja, Y. and Modi, K. *Cloud computing-concepts, architecture and challenges*. in *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*. 2012. IEEE.

197. Tsai, W., Bai, X. and Huang, Y. *Software-as-a-service (SaaS): perspectives and challenges.* Science China Information Sciences, 2014. **57**(5): p. 1-15.

198. Kolb, S. and Wirtz, G. *Towards application portability in platform as a service*. in *2014 IEEE 8th International Symposium on Service Oriented System Engineering*. 2014. IEEE.

199. Kächele, S., et al. *Beyond IaaS and PaaS: An extended cloud taxonomy for computation, storage and networking*. in *2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing*. 2013. IEEE.

200. Kumar, S. and Goudar, R. *Cloud computing-research issues, challenges, architecture, platforms and applications: a survey.* International Journal of Future Computer and Communication, 2012. **1**(4): p. 356.

201. https://www.bmc.com/blogs/saas-vs-paas-vs-iaas-whats-the-difference-and-how-to-choose/.

202. https://effectivesoftwaredesign.com/2014/11/20/conference-talk-hayim-makabee-on-software-quality-attributes/.

203. Yahaya, J.H. and Deraman, A. *Measuring unmeasurable attributes of software quality using pragmatic quality factor*. in *2010 3rd International Conference on Computer Science and Information Technology*. 2010. IEEE.

204. Gaffney Jr, J.E. *Metrics in software quality assurance*. in *Proceedings of the ACM'81 conference*. 1981. ACM.

205. Vollman, T.E., *Software quality assessment and standards.* Computer, 1993. **26**(6): p. 118-120.

206. Garraghan, P., Townend, P. and Xu, J. *An analysis of the server characteristics and resource utilization in google cloud*. in *2013 IEEE International Conference on Cloud Engineering (IC2E)*. 2013. IEEE.

207. Carvalho, M., Menascé, D.A. and Brasileiro, F. *Capacity planning for IaaS cloud providers offering multiple service classes.* Future Generation Computer Systems, 2017. **77**: p. 97-111.

208. Galante, G., et al., *An analysis of public clouds elasticity in the execution of scientific applications: a survey.* Journal of Grid Computing, 2016. **14**(2): p. 193-216.

209. Al-Dhuraibi, Y., et al., *Elasticity in cloud computing: state of the art and research challenges.* IEEE Transactions on Services Computing, 2017. **11**(2): p. 430-447.

210. Tchernykh, A., et al., *Towards understanding uncertainty in cloud computing with risks of confidentiality, integrity, and availability.* Journal of Computational Science, 2016.

211. Habib, S.M., Ries, S. and Muhlhauser, M. *Cloud computing landscape and research challenges regarding trust and reputation*. in *2010 7th International Conference on*

Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing. 2010. IEEE.

212. Sun, Y., et al., *Data security and privacy in cloud computing.* International Journal of Distributed Sensor Networks, 2014. **10**(7): p. 190903.

213. Modi, C.N. and Acha, K. *Virtualization layer security challenges and intrusion detection/prevention systems in cloud computing: a comprehensive review.* the Journal of Supercomputing, 2017. **73**(3): p. 1192-1234.

214. Vaquero, L.M., Rodero-Merino, L. and Morán, D. *Locking the sky: a survey on IaaS cloud security.* Computing, 2011. **91**(1): p. 93-118.

215. Wu, W., et al., *How to achieve non-repudiation of origin with privacy protection in cloud computing.* Journal of Computer and System Sciences, 2013. **79**(8): p. 1200-1213.

216. Zhou, J. and Gollmann, D. *Evidence and non-repudiation.* Journal of Network and Computer Applications, 1997. **20**(3): p. 267-281.

217. Ahmad, I., Bakht, H. and Mohan, U. *Cloud Computing–Threats and Challenges.* J. Comput. Manag. Stud, 2017. **1**(1).

218. Zhou, J., *Non-repudiation in electronic commerce*. 2001: Artech House.

219. Contractor, D. and Patel, D.R. *Accountability in Cloud Computing by Means of Chain of Trust.* IJ Network Security, 2017. **19**(2): p. 251-259.

220. Jaatun, M.G., et al., *Enhancing accountability in the cloud.* International Journal of Information Management, 2016.

221. Jaatun, M.G., et al. *Accountability requirements for the cloud*. in *2017 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. 2017. IEEE.

222. Upadhyaya, A. and Bansal, M. *Deployment of secure sharing: Authenticity and authorization using cryptography in cloud environment*. in *2015 International Conference on Advances in Computer Engineering and Applications*. 2015. IEEE.

223. Indu, I., Anand, P.R. and Bhaskar, V. *Identity and access management in cloud environment: Mechanisms and challenges.* Engineering science and technology, an international journal, 2018. **21**(4): p. 574-588.

224. Mesbahi, M.R., Rahmani, A.M. and Hosseinzadeh, M. *Reliability and high availability in cloud computing environments: a reference roadmap.* Human-centric Computing and Information Sciences, 2018. **8**(1): p. 20.

225. Kumari, P. and Kaur, P. *A survey of fault tolerance in cloud computing.* Journal of King Saud University-Computer and Information Sciences, 2018.

226. Amin, Z., Singh, H. and Sethi, N. *Review on fault tolerance techniques in cloud computing.* International Journal of Computer Applications, 2015. **116**(18).

227. Cheraghlou, M.N., Khadem-Zadeh, A. and Haghparast, M. *A survey of fault tolerance architecture in cloud computing.* Journal of Network and Computer Applications, 2016. **61**: p. 81-92.

228. Essa, Y.M., *A survey of cloud computing fault tolerance: Techniques and implementation.* International Journal of Computer Applications, 2016. **138**(13).

229. Saikia, L.P. and Devi, Y.L. *Fault tolerance techniques and algorithms in cloud computing.* International Journal of Computer Science & Communication Networks, 2014. **4**(1): p. 01-08.

230. Singh, G. and Kinger, S. *A survey on fault tolerance techniques and methods in cloud computing.* International Journal of Engineering Research and Technology, 2013. **2**(6).

231. Prathiba, S. and Sowvarnica, S. *Survey of failures and fault tolerance in cloud*. in *2017 2nd International Conference on Computing and Communications Technologies (ICCCT)*. 2017. IEEE.

232. ISO/IEC, *25021 (2012) Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measure elements.* International Organization for Standardization, Geneva, Switzerland, 2012.

233. Azarnik, A. and Shayan, J. *Associated risks of cloud computing for SMEs.* Open International Journal of Informatics (OIJI), 2012. **1**(1): p. 37-45.

234. Alam, M.G.R., et al. *Cloud based mental state monitoring system for suicide risk reconnaissance using wearable bio-sensors*. in *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*. 2014. ACM.

235. Luo, L., et al., *Simulation of power consumption of cloud data centers.* Simulation Modelling Practice and Theory, 2013. **39**: p. 152-171.

236. Ghafari, S.M., et al. *Bee-MMT: A load balancing method for power consumption management in cloud computing*. in *2013 Sixth International Conference on Contemporary Computing (IC3)*. 2013. IEEE.

237. Bui, D.-M., et al., *Energy efficiency for cloud computing system based on predictive optimization.* Journal of Parallel and Distributed Computing, 2017. **102**: p. 103-114.

238. Dalapati, P. and Sahoo, G. *Green solution for cloud computing with load balancing and power consumption management.* Int. J. Emerg. Technol. Adv. Eng, 2013. **3**(3): p. 353-359.

239. Kaplan, J.M., Forrest, W. and Kindler, N. *Revolutionizing data center energy efficiency*. 2008, Technical report, McKinsey & Company.

240. https://finance.yahoo.com/news/every-google-search-results-co2-090042259.html.

241. Roy, S., Pattnaik, P.K. and Mall, R. *A cognitive approach for evaluating the usability of Storage as a Service in Cloud Computing Environment.* International Journal of Electrical & Computer Engineering (2088-8708), 2016. **6**(2).

242. Condori-Fernandez, N. and Lago, P. *Characterizing the contribution of quality requirements to software sustainability.* Journal of Systems and Software, 2018. **137**: p. 289-305.

243. Gordieiev, O., Kharchenko, V.S. and Vereshchak, K. *Usable Security Versus Secure Usability: an Assessment of Attributes Interaction*. in *ICTERI*. 2017.

244. Zhou, P., et al. *Quality Model of Cloud Service*. in *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems*. 2015. IEEE.

245. Norman, D., *The design of everyday things: Revised and expanded edition*. 2013: Basic books.

246. Maity, R. and Bhattacharya, S. *A model to compute webpage aesthetics quality based on wireframe geometry*. in *IFIP Conference on Human-Computer Interaction*. 2017. Springer.

247. Ngo, D.C.L., Teo, L.S. and Byrne, J.G. *Modelling interface aesthetics.* Information Sciences, 2003. **152**: p. 25-46.

248. Ray, D., *Cloud adoption decisions: Benefitting from an integrated perspective.* Electronic Journal Information Systems Evaluation, 2016. **19**(1).

249. Saya, S., Pee, L.G. and Kankanhalli, A. *The Impact of Institutional Influences on Perceived Technological Characteristics and Real Options in Cloud Computing Adoption*. in *ICIS*. 2010.

250. Birman, K., Chockler, G. and van Renesse, R. *Toward a cloud computing research agenda.* ACM SIGACt News, 2009. **40**(2): p. 68-80.

251. Harfoushi, O., et al., *Factors affecting the intention of adopting cloud computing in Jordanian hospitals.* Communications and Network, 2016. **8**(02): p. 88.

252. Tweel, A., *Examining the relationship between technological, organizational, and environmental factors and cloud computing adoption*. 2012: Northcentral University.

253. Panetto, H., et al., *New perspectives for the future interoperable enterprise systems.* Computers in Industry, 2016. **79**: p. 47-63.

254. Lewis, G.A. *Role of standards in cloud-computing interoperability*. in *2013 46th Hawaii International Conference on System Sciences*. 2013. IEEE.

255. Mansour, I., et al. *Interoperability in the heterogeneous cloud environment: a survey of recent user-centric approaches*. in *Proceedings of the International Conference on Internet of things and Cloud Computing*. 2016. ACM.

256. Mourad, M., Nassehi, A. and Schaefer, D. *Interoperability as a Key Enabler for Manufacturing in the Cloud.* Procedia CIRP, 2016. **52**: p. 30-34.

257. Haile, N. and Altmann, J. *Evaluating investments in portability and interoperability between software service platforms.* Future Generation Computer Systems, 2018. **78**: p. 224-241.

258. Moreno-Vozmediano, R., Montero, R.S. and Llorente, I.M. *Key challenges in cloud computing: Enabling the future internet of services.* IEEE Internet Computing, 2012. **17**(4): p. 18-25.

259. Vuyyuru, M., et al., *An overview of cloud computing technology.* International Journal of Soft Computing and Engineering, 2012. **2**(3): p. 244.

260. Casalicchio, E. and Silvestri, L. *Mechanisms for SLA provisioning in cloud-based service providers.* Computer Networks, 2013. **57**(3): p. 795-810.

261. Ashraf, I., *An overview of service models of cloud computing.* International Journal of Multidisciplinary and Current Research, 2014. **2**(1): p. 779-783.

262. Lee, J.Y., Lee, J.W. and Kim, S.D. *A quality model for evaluating software-as-a-service in cloud computing*. in *2009 seventh ACIS international conference on software engineering research, management and applications*. 2009. IEEE.

263. Vafaei, N., Ribeiro, R.A. and Camarinha-Matos, L.M. *Normalization techniques for multi-criteria decision making: analytical hierarchy process case study*. in *doctoral conference on computing, electrical and industrial systems*. 2016. Springer.

264. Vinogradova, I., Podvezko, V. and Zavadskas, E. *The recalculation of the weights of criteria in MCDM methods using the bayes approach.* Symmetry, 2018. **10**(6): p. 205.

265. Jin, Z. and Chen, Y. *Telemedicine in the cloud era: Prospects and challenges.* IEEE Pervasive Computing, 2015. **14**(1): p. 54-61.

266. John, N. and Shenoy, S. *Health cloud-Healthcare as a service (HaaS)*. in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. 2014. IEEE.

267. https://www.itnonline.com/article/ins-and-outs-cloud-based-pacs.

268. Silva, L.A.B., Costa, C. and Oliveira, J.L. *A PACS archive architecture supported on cloud services.* International journal of computer assisted radiology and surgery, 2012. **7**(3): p. 349-358.

269. Lorido-Botran, T., Miguel-Alonso, J. and Lozano, J.A. *A review of auto-scaling techniques for elastic applications in cloud environments.* Journal of grid computing, 2014. **12**(4): p. 559-592.

270. Kaur, R. and Luthra, P. *Load balancing in cloud computing*. in *Proceedings of international conference on recent trends in information, telecommunication and computing, ITC*. 2012. Citeseer.

271. Mao, M. and Humphrey, M. *A performance study on the vm startup time in the cloud*. in *2012 IEEE Fifth International Conference on Cloud Computing*. 2012. IEEE.

272. Amrhein, D., Anderson, P. and de Andrade, A. *Cloud computing use cases. A white paper produced by the Cloud Computing Use Case Discussion Group*. 2009, Tech. Rep. Version, 2.