The ACCQ Methodology: Assuring Consumer Satisfaction of Cloud Service Quality

A Bachelor's Thesis in Information Sciences

July 13, 2013

Author:
Joost Janssen

Student I.D:
0709387

Supervisor:
Dr. Peter Achten

Institution:
Radboud University,
Nijmegen (NL)

Abstract

Cloud Computing offers virtualized, dynamic and scalable IT technologies on demand in the form of cloud services. While methods exist that are capable of assuring the quality of cloud services consumed, consumer satisfaction of a cloud service is influenced by more aspects than service quality alone. In order to retain customers, cloud service providers will need to assure consumer satisfaction of their services' quality. Towards this goal, this thesis proposes the ACCQ-methodology. This two-step methodology prescribes how to determine which aspects of cloud service quality influence consumer satisfaction during service consumption, as well as what capabilities are required of a service provider or assurance method for cloud service quality satisfaction assurance. The ACCQ-methodology can be used to assess existing assurance methods in their capability of assuring consumer satisfaction, or it can be used as a framework upon which a method for assuring consumer satisfaction of a specific cloud service's quality can be developed. An application of the ACCQ-methodology on the DYNAMICO assurance framework demonstrates its use in assessing existing methods.

Contents

Chapter 1

Introduction

Increasing numbers of users and businesses are migrating towards the use of the relatively new Cloud Computing paradigm. The main distinction between Cloud Computing and more "traditional" ICT technologies lies in the virtualized character of Cloud Computing. As opposed to conventional ICT, where all hard- and software used is physically present on-premises, Cloud Computing resources are essentially "rented" on demand and accessed remotely. This yields Cloud Computing a service-like nature, hence it being often referred to as cloud services. The main advantage of consuming IT resources in a service-like setting is that consumers no longer have to invest in for example expensive infrastructures while still being able to benefit from the advantages of having access to such infrastructures; consumers can conveniently pay for the use of such infrastructures for merely the duration and scope of the resources.

While methods exist that attempt to deal with dynamic service configuration and quality assurance, these seem to be developed with the service provider's best interests at heart rather than the consumer's. This becomes apparent from the emphasis on SLA violation prevention in order to avoid service providers having to pay violation fees. Service consumers' interests in dynamic service configuration and quality assurance likely exceed the requirements as put forth in Service Level Agreements, as the consumer likely expects service configuration and quality to uphold to a satisfactory degree in situations such as service consumption environment fluctuations resulting from for example business rules adaptation or changes in cloud service requirements during service consumption.

In this thesis, Cloud Computing and Consumer Satisfaction theories are examined and compared in order to determine which factors of cloud service quality can influence consumer satisfaction. A methodology is developed that prescribes

- how to determine which factors of cloud service quality can influence consumer satisfaction of cloud service quality during service consumption, as well as
- what is required of a service provider or satisfaction assurance method to assure consumer satisfaction during service consumption.

This methodology can be used either to assess existing assurance methods in their capability to ensure consumer satisfaction of cloud service quality in general or for a specific cloud service, or it can be used as a reference model to support the development of a method to assure consumer satisfaction of (a specific) cloud service(s) during service consumption.

1.1 Research Method

Research Goal

The goal of this research is to develop a methodology prescribing *how* to determine which factors of cloud service quality can influence consumer satisfaction during service consumption and *what* is required of a service provider or satisfaction assurance method to assure consumer satisfaction of cloud service quality during service consumption.

Research Questions

1. Which aspects of cloud services can influence consumer satisfaction of cloud service quality? (Theoretical Context)
   (a) Which aspects of cloud services can be used to judge cloud service quality? (Cloud Computing theory)
   (b) Which aspects of services (in general) can influence consumer satisfaction? (Consumer Satisfaction theory)
   (c) Which aspects of cloud service quality can influence consumer satisfaction of cloud service quality?
2. What capabilities are required of a service provider or assurance method to assure consumer satisfaction of cloud service quality *during service consumption*?
   (a) Which aspects of cloud service quality can influence consumer satisfaction of cloud service quality *during service consumption*?
   (b) What capabilities are required to assure consumer satisfaction of cloud service quality *during service consumption*?

Research Methodology

An analysis of consumer satisfaction in respect of cloud service quality results in a set of quality attributes influencing consumer satisfaction. A methodology will be developed which prescribes how to determine which of these quality attributes influence consumer satisfaction *during service consumption*. Furthermore, this methodology will prescribe what capabilities are required of a service provider or an assurance method in order to be able to assure consumer satisfaction of cloud service quality during service consumption. Subsequently, this methodology will be applied on the DYNAMICO quality assurance framework [VTM+13] in order to demonstrate how to assess an assurance method on its capability to assure consumer satisfaction of cloud service quality. The development of the consumer satisfaction assurance methodology and the application of this methodology on DYNAMICO will result in a set of conclusions regarding assurance of consumer satisfaction of cloud service quality, as depicted in Figure 1.

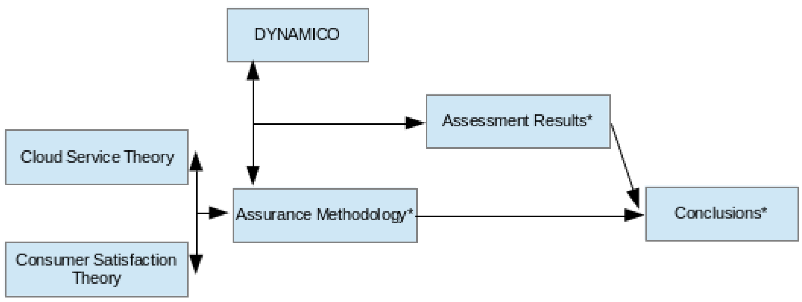

Figure 1: Research Method (* represent original contributions).

1.2 Roadmap

This thesis is structured as follows. Chapter 2 briefly outlines Parasuraman et al.'s “classical” SERVQUAL instrument for measuring service quality [PZB88]. Chapter 3 examines the Cloud Computing paradigm and cloud service quality. Chapter 4 explores the role of consumer satisfaction in services in general.

In Chapter 5, the ACCQ-methodology concerning assuring consumer satisfaction of cloud service quality during service consumption is developed, based on the theoretical context discussed in Chapters 3 and 4. Chapter 6 demonstrates the application of the ACCQ-methodology towards assessing existing assurance methods by applying it to Villegas et al.'s DYNAMICO assurance framework [VTM+13].

Chapter 7 provides an overview of research closely related to the research conducted in this thesis. Chapter 8 presents a summary and conclusion of this thesis.

Chapter 2

Theoretical Context: SERVQUAL

In 1988, Parasuraman et al. [PZB88] developed the SERVQUAL (SERVice QUALity) instrument for measuring consumer perceptions of service quality in service and retailing organizations. Over the years, this instrument has become widely accepted to serve as a basis for research in consumer perception of service quality. Furthermore, this theory has been extended in many directions, amongst which electronic service quality (E-S-QUAL [PZM05]) and Software-as-a-Service quality (SaaSQual [BKH11]). Because SERVQUAL plays such a fundamental role in both (electronic) service quality and consumer satisfaction theories, this chapter will provide a short introduction to SERVQUAL before exploring the fields of cloud service quality (Chapter 3) and consumer satisfaction (Chapter 4).

2.1 Purpose

Due to features of intangibility, heterogeneity and inseparability of production and consumption, objective measures of product quality such as durability or number of defects are non-existent in services [PZB88, 13]. To compensate, service quality can be assessed by measuring consumers' perceptions of quality [PZB88, 13]. Parasuraman et al. have developed the SERVQUAL scale to quantify these perceptions.

2.2 Concepts

SERVQUAL is based on *perceived quality*, "the consumer's judgment about an entity's overall excellence or superiority [...]; it is a form of attitude [...] and results from a comparison of expectations with perceptions of performance". [PZB88, 15]. In exploratory research [PZB85], Parasuraman et al. have distinguished ten service quality dimensions by which consumers assess service quality which served as a basis for SERVQUAL scale item generation. After empirical research on the ability of these dimensions to discriminate amongst consumer's perceptions of quality and purifying / condensing the dimensions accordingly, the following 5 dimensions of SERVQUAL were defined [PZB88, 23]:

Tangibles: Physical facilities, equipment, and appearance of personnel.

Reliability: Ability to perform the promised service dependably and accurately.

Responsiveness: Willingness to help customers and provide prompt service.

Assurance: Knowledge and courtesy of employees and their ability to inspire trust and confidence.

Empathy: Caring, individualized attention the firm provides its customers.

Each of these dimensions is comprised of 4 or 5 scale items (22 items in total), each indicating a perspective of its quality dimension (see Appendix A). The SERVQUAL instrument poses two questions to service consumers on each of these items; it surveys the consumer's *expected* quality of a service's features ("To what extent do you [the consumer] think [a specific] service should possess the [following] features?"), as well as the consumer's *perceived* quality of a service's features ("To what extent do you [the consumer] believe [a specific service] has the [following] features?") [PZB88].

2.3 Applications

Parasuraman et al. indicate several examplatory applications of the SERVQUAL scale [PZB88]:

- Periodically tracking service quality trends;
- Assessing a firm's (service) quality along each quality dimension, or overall quality by averaging over dimensions. This is limited to quality perceived by current or past customers of the firm;
- Determining the relative importance of a quality dimension within customers' overall quality perceptions;

- Comparing service performance to competitors;
- Categorizing consumers into perceived-quality segments based on individual SERVQUAL scores (see section 2.4).

2.4 Perceived-quality Segments

By categorizing a firm's consumers into several perceived-quality segments, for example "high", "medium" or "low" quality perceived by the consumer, Parasuraman et al. suggest these segments can be further analyzed in order to gain insight on how a firm can improve service quality in the eyes of important customer groups. This analysis can be based on [PZB88, 35]

- Demographic, psychographic and/or other profiles;
- The relative importance of the five quality dimensions influencing service quality perception; and
- Reasons behind the perceptions reported.

The relative importance of quality dimensions is measured by comparing accumulated "perception-expectation gap scores" of each dimension. Gap scores indicate the difference between a consumer's *expected* and *perceived* quality of a service's feature as indicated in the SERVQUAL instrument [PZB88, 35].

Chapter 3

Theoretical Context: Cloud Computing

Cloud Computing presents a relatively new approach to consuming and providing software as a service rather than a good, as has long been the convention. This chapter explores the basics of Cloud Computing and cloud service quality. Types of cloud services, a cloud service's lifetime cycle, definitions of cloud service quality and sets of service quality attributes are introduced in this chapter.

3.1 Cloud Services

3.1.1 Types of Cloud Services

There are three basic tiers of cloud services, each tier consisting of technologies which provide support for the tiers laying above [HZ13, 578] and can be provided "as a Service" separately or in combination:

- Infrastructure as a Service (IaaS) offers "processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software [including] operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications" [MG12, 3]. Popular examples of IaaS are Amazon EC2 [Zhu10, 21] or the open-source Eucalyptus [Sos11].
- Platform as a Service (PaaS) offers "[t]he capability [...] to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools sup-

ported by the [service] provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications" [MG12, 2]. Popular examples of PaaS are Google AppEngine or Microsoft Azure [JASZ11, 3][Zhu10, 22].

- Software as a Service (SaaS) offers "[t]he capability [...] to use the [service] provider's applications running on a cloud infrastructure. [...] The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings" [MG12, 2]. Popular examples of SaaS are Google Apps [JASZ11, 3] such as Google Docs or Salesforce [LKN+09, 31].

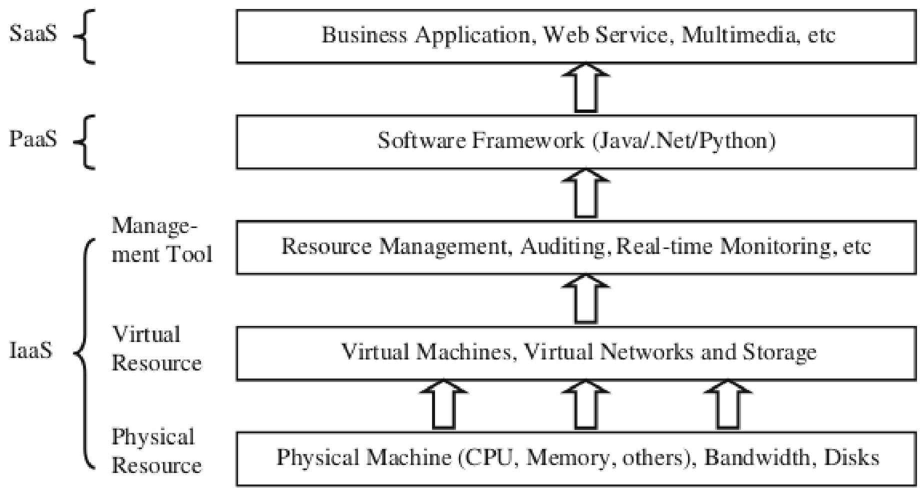

Figure 2: Cloud Service Tiers [HZ13, JASZ11].

The correspondence between these cloud service technologies is that they all offer services where the consumer has virtualized and abstracted access to the necessary resources through standard networking protocols [Sos11, 3], while never having direct contact with the physical resources used. The U.S. National Institute of Standards and Technology has identified 5 essential characteristics of cloud computing [MG12]:

- On-demand self-service: a customer should be able to request cloud services as needed, without human interaction;

- Broad network access: network access should be through standard mechanisms supported by multiple platforms (mobile phones as well as superservers);

- Resource pooling: the service provider's resources are pooled to serve multiple users on-demand; a user is not aware of the (location of the) physical resources being used, creating some degree of location independence;
- Rapid elasticity: resources should be able to be provisioned and released in an elastic, sometimes automatic, manner in order to provide for rapid scaling of capabilities;
- Measured service: resource usage should be monitored, controlled and reported to provide transparency and a means for compensation for services offered, typically on a pay/charge-per-use basis.

3.1.2 Cloud Service Lifetime Cycle

Cloud service consumption is a dynamic process in which both the consumer and the service provider play a continuous role. This section explores several definitions of this process that have been developed.

ITIL V3 Service Lifecycle

The United Kingdom's Office of Government Commerce (OCG) has derived a set of best practices in IT service management from both the public and private sectors, collected in an Information Technology Infrastructure Library (ITIL), organized around the following Service Lifecycle [Arr10, 3], depicted in Figure 3:

- Service Strategy;
- Service Design;
- Service Transition;
- Service Operation; and
- Continual Service Improvement.

In the *Service Strategy* phase, a consumer defines the market space for the planned service(s), sets the service's performance expectations, identifies, prioritizes and selects opportunities, and develops policies, guidelines and processes to be used to manage the service(s) [Com08, 12]. From the

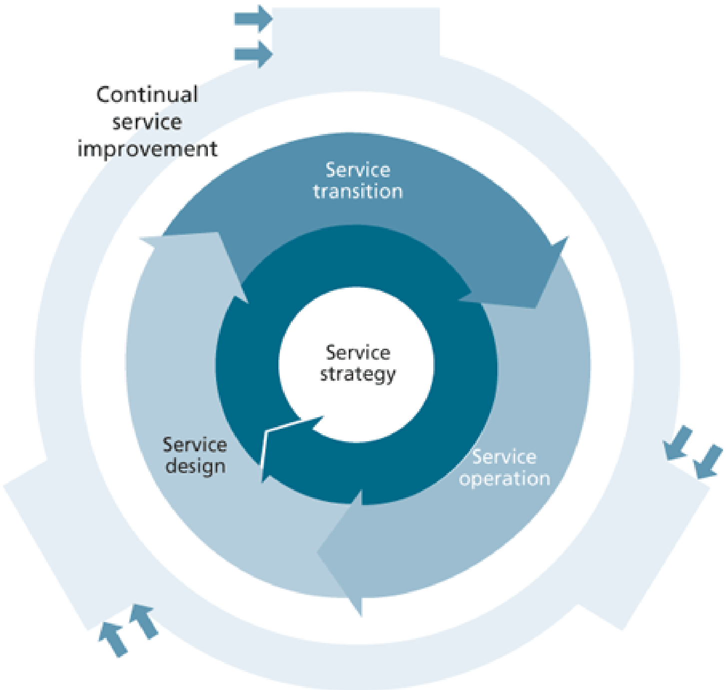

Figure 3: ITIL V3 Service Lifecycle [CHR+07].

provider's perspective, the consumers and their needs are identified, the capabilities and resources required to meet these needs are determined and the requirements for successful service execution are set up [Arr10, 3].

The *Service Design* phase "assures that [...] services are designed effectively to meet customer expectations" [Arr10, 3]. This entails that the service provider accumulates, designs and develops new and changed services that meet the consumer's business requirements as set forth in the Service Strategy phase, as well as processes that govern the management and delivery of services [Com08, 14].

In the *Service Transition* phase, the "new or changed services developed in the Service Design phase are transitioned into [the Service Operation phase] while controlling the risks of service failure and business disruption" [Com08, 17]. This is realized by "controlling the assets and configuration items (underlying components [such as] hardware [and] software) [...], service validation and testing and transition planning to ensure that users, support personnel and the production environment have been prepared for the release to production" [Arr10, 3-4].

During the *Service Operation* phase, the service is then delivered "on an ongoing basis, overseeing the overall health of the service[,] includ[ing] managing disruptions to service through rapid restoration of incidents, determining the root cause of problems and detecting trends associated with recurring issues, handling daily routine end[-]user requests and managing service access" [Arr10, 4].

Throughout the service's lifecycle, *Continual Service Improvement* "offers a mechanism for IT to measure and improve the service levels, the technology and the efficiency and effectiveness of processes used in the overall management of services [Arr10, 4].

Joshi et al.'s Integrated Service Lifecycle

An academic approach to describing a Cloud service's lifecycle has been proposed by Joshi et al. [JFY09]. In their model, the IT cloud service lifecycle has been divided into five phases, sequentially

1. Requirements Specification;
2. Service Discovery;
3. Service Negotiation;
4. Service Composition; and
5. Service Consumption.

Each of these phases contains subphases, as shown in Figure 4. The phases in Figure 4 are colorcoded as described below.

During the *Requirements Specification* phase (blue), the consumer identifies the domain as well as the technical, functional and non-functional specifications of the service to be consumed.

In the *Service Discovery* phase (green), the consumer issues a request for service either to one or multiple potential primary service provider(s) or to a service discovery engine, detailing the requirements gathered in the first phase. The discovery phase produces a (ranked) list of services or service combinations which satisfy as much of the requirements as achievable.

In the next phase, *Service Negotiation* (purple), the consumer and the potential primary service provider negotiate on the service to be delivered and its acceptance criteria and record this in a Service Level Agreement or SLA (see Section 3.2.1).

In the *Service Composition* phase (magenta), the primary service provider bundles the services discovered and selected in the previous phases into a single service to be offered to the consumer. The consumer usually does not (have to) notice whether a (sub)service is offered by their primary service provider or by a third-party / secondary provider.

During the final phase of *Service Consumption* (pink), the service is delivered to the consumer, the consumer pays for the services consumed and

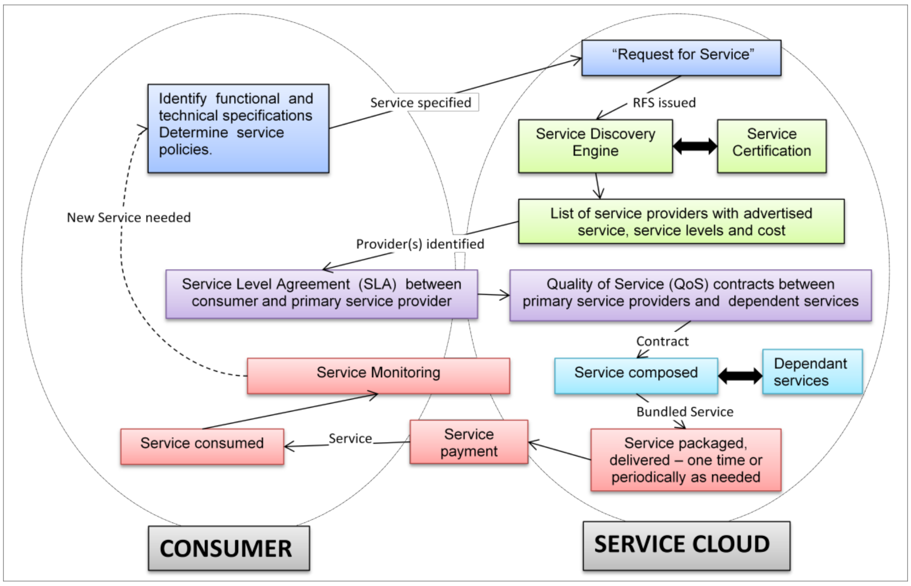

Figure 4: Joshi et al.'s Integrated Service Lifecycle [JFY09].

service quality is monitored to check if the Service Level Agreement is being complied to [JFY09].

A major difference between Joshi et al.'s lifecycle model and the one presented in ITIL V3 is the lack of a continual improvement-like feature in the former model. Joshi et al.'s model implies the lifecycle is cyclical only through the consumer's recognition of a "New Service needed" (see Figure 4), while service adaptation is not included in their model. While Service Monitoring is part of Joshi et al.'s lifecycle's Service Consumption phase, the lifecycle does not provide for feeding back Service Monitoring results into the consumed service. Nonetheless, their Integrated Lifecycle model does provide clear insight into the relationship between a service provider and the consumer. Joshi et al. do propose a set of *Service Metrics* used for "track[ing] performance of each phase of the lifecycle[,] ensuring successful discovery, composition and consumption of the services" [JFY09, 3] during the Service Monitoring subphase. These and other metrics are elaborated on in Section 3.2.2.

3.2 Cloud Service Quality

In the following section various methods of measuring cloud service quality or performance are discussed. Different sets of performance metrics, both from the provider's and consumer's perspective, are introduced.

3.2.1 Service Level Agreement

Typically, cloud service consumers and providers compose a contract detailing the service to be delivered and its acceptance criteria. This contract, or Service Level Agreement (SLA), commonly specifies at least the following parameters: availability of the service (uptime), response times (latency), reliability of the service components, responsibilities of each party involved, delivery mode, service cost and warranties to be issued [Sos11, 39-40], [JFY09, 3]. Nowadays, most cloud service SLAs are standardized as often (nearly) identical services are being provided by a single company for many different customers; only in cases of (heavily) customized services or a single client becoming a large consumer of services [Sos11, 39] custom SLAs are negotiated between consumer and service provider. Additionally, in the case of the primary service provider planning on integrating services from secondary service providers as components in a bundled service, Quality of Service agreements between the primary and secondary providers are negotiated in order to warrant the primary provider's capability to fulfill the Service Level Agreement. Quality of Service, or QoS, is a collection of technical properties of a service, including availability, security, response time and throughput [Men02, 1], mainly focusing on network performance. As apparent from the SLA parameters above and Sosinsky's definition of an SLA as a "contract for performance negotiated between [the consumer] and a service provider" [Sos11, 39], these agreements tend to focus on service *performance*. SLAs are agreed upon during the Service Design and/or Negotiation phases, while the service's actual performance is compared to the performance as agreed upon in the SLA throughout the Service Consumption phase by means of service monitoring techniques. Both the consumer and the service provider have an interest in monitoring service quality: the consumer needs to be assured they receive the service they pay for, while the service provider needs to verify it meets its contractual obligations. Violations of SLA parameters often result in "[the provider being] punished by having to offer the client a credit or pay a penalty" [Sos11, 40].

3.2.2 Cloud Service Quality Attributes

In the following section, sets of attributes indicative of cloud service quality from various sources are accumulated and documented. The resulting quality attributes will be used to examine which aspects of cloud service quality influence consumer satisfaction in later chapters.

Joshi et al.'s Service Lifecycle Quality Metrics

In developing their Integrated Service Lifecycle, Joshi et al. [JFY09] have identified a set of service metrics tracking service performance during each phase of the service lifecycle (see Section 3.1.2). In their article presenting the Integrated Lifecycle, they present some “key metrics that should be tracked to ensure high service quality” [JFY09, 3], shown in Table 1.

Quality Metrics	**Phase**	**Definition**
Data quality	Requirements, Discovery	The quality of data delivered by the service.
Cost	Requirements, Discovery, Consumption	Costs of the service for the consumer.
Security	Requirements, Discovery	Required security / permission levels of the service.
Service Gap	Discovery	The gaps that exist between the consumer's requirements and the functionalities of services available off the shelf.
Certificate	Discovery	Certification of the service provider to be able to meet service requirements and constraints. Issued by an independent body.
SLA	Negotiation, Consumption	Service Level Agreement between consumer and primary provider. Includes security policy and data quality policy.
QoS	Negotiation, Consumption	Quality of service agreement between primary provider and component providers.
Delivery mode	Consumption	Service delivered in real-time, batch mode or as a one-time service.
Payment options	Negotiation, Consumption	Service payment will be up-front or on a periodic basis (monthly, quarterly, annual etc.). Depending on the option selected, the service will be delivered before or after payment.

Coupling	Composition	Coupling determines how dependent the service is on other services or resources for its delivery.
Cohesion	Composition	Cohesion measures the extent to which related aspects of a requirement are kept together in the same service, and unrelated aspects are kept out.
Reliability	Consumption	Reliability tracks the service quality to ensure the service functionality and data accuracy is maintained.
Performance	Consumption	Tracks the service performance, including throughput, latency and response time.
Consumer Satisfaction	Consumption	Periodically the provider tracks (via surveys, opinion polls etc.) whether consumers are satisfied with the service.

Table 1: Joshi et al.'s Service Quality Metrics [JFY09].

Hu and Zhang's Evaluation System for Cloud Service Quality based on SERVQUAL

Hu and Zhang [HZ13] have developed an evaluation system for cloud service quality based on SERVQUAL, as shown in Table 2. They have derived this system from Benlian et al.'s SaaS-QuaL (Software-as-a-Service Quality) measure [BKH11], extracting the following service quality dimensions [HZ13]:

- *Rapport*: Service quality of technology support and customer care made available by cloud service providers;
- *Responsiveness*: Capacity of service providers to ensure service availability and normal performance;
- *Reliability*: Capacity of service providers to provide the cloud service correctly and in time;
- *Flexibility*: Capacity of service providers to support users to change the flexibility of default parameters; and

- *Security*: Systematic protection methods adopted by service providers to avoid data missing and system collapse.

Hu and Zhang [HZ13] have omitted Benlian et al.'s [BKH11] *Features* quality dimension, which "describe[s] perceptive specifics and function. [It was omitted] because it can't be measured by external metrics" [HZ13, 580]. Benlian et al. have defined the *Features* dimension as follows:

- *Features*: "the degree the key functionalities (e.g., data extraction, reporting, or configuration features) and design features (e.g., user interface) of an SaaS application meet the business requirements of a customer" [BKH11, 99].

Even though external or empirical metrics for the measurement of quality dimension *Features* have not been defined by either Hu and Zhang or Benlian et al., they may be derived by comparing its indicators to the consumer's business requirements. Table 3 displays the *Features* quality dimension indicators as defined by Benlian et al. [BKH11].

Quality Dimension	Indicators	Empirical Variables
Rapport	Training system	Training hours
	Customized service	Personalized information pushing frequency
	Customer support	SLA selection performance
Responsiveness	Dynamic scalability response	Response time
	Disaster recovery time	Response time
	Technology support availability	Non-support times during commitment
Reliability	Elastic service availability	Available uptime percentage
	Service accuracy	Failure frequency
	Budget control	SLA selection performance
Flexibility	Multi-client access adjustment	Multi-client variation balance
	Extra resources allocation	Coverage of resources (in IaaS)
	Data migration	Amount of data migration (in PaaS)
Security	Data backup	Image backup frequency (in IaaS)
		Database backup frequency (in PaaS)
	Fault recovery strategy	SLA selection performance
	Regular security audit	SLA selection performance
	Anti-virus tool	SLA selection performance
	Data secrecy	SLA selection performance
	Access control	SLA selection performance

Table 2: Hu and Zhang's Cloud Service Quality Evaluation System based on SERVQUAL [HZ13].

Quality Dimension	Indicators
Features	Visually appealing, sympathetic user interface
	User-friendly navigation structure, search functionality
	Data reporting, extracting features
	Configuration features
	Help functionalities
	Dashboard features with customer's service usage metrics
	Core features supporting process steps / activities

Table 3: Benlian et al.'s *Features* quality dimension [BKH11].

3.2.3 Service Quality from the Provider's Perspective

In order to prevent penalties due to SLA violations, service providers continuously need to monitor service performance metrics and compare these to the lower performance limits specified in the SLA, thereby safeguarding contractual service quality obligations from the provider's perspective. Additionally, in cases of bundled services, subservice Quality of Service (QoS) performance and dependent services' performance needs to be monitored (see Figure 2). Ample SLA- and QoS-monitoring mechanisms to be incorporated into provisioned cloud services have been developed to meet this requirement:

- Patel et al. [PRS09] have developed a mechanism for managing SLAs in a cloud computing environment using the Web Service Level Agreement (WSLA) framework.
- Ferretti et al. [FGP+10] have developed a middleware architecture designed to respond effectively to QoS requirements by reconfiguration in cases of SLA being violated or honored.
- Maurer et al. [MBS12] have devised a self-adaptive approach balancing minimization of SLA violations, maximization of resource utilization and minimization reconfiguration actions.
- Sakr and Liu [SL12] present a framework facilitating adaptive and dynamic service provisioning based on application-defined policies for satisfying SLA performance requirements.
- Freitas et al. [FPP12] propose an approach for specifying and enforcing SLAs, including performance and fault-tolerance QoS assurance

mechanisms (see Section 3.3).

3.2.4 Service Quality from the Consumer's Perspective

From the consumer's perspective, service quality exceeds SLA compliance; consumers may have additional expectations of a high-quality service's performance. Koehler et al. argue that "[p]revious research often named technical issues[, ... while] lack[ing] an explanation for the impact on consumer preferences by improving [...] technical problems" [KMA10, 3]. They try to overcome this issue by identifying consumer preferences for cloud service attributes by ways of a survey in small and medium enterprises. Suitable attributes were selected through document review and expert interviews, resulting in the following attributes [KMA10, 6]:

- Provider Reputation;
- Required Skills;
- Migration Process;
- Pricing Tariff;
- Cost compared to intern[al] solution; and
- Consumer Support.

The relative importance levels of these attributes derived from the survey's answers are shown in Table 4. A higher relative importance level means this attribute weighs heavier in consumer's preferences than an attribute with a lower importance level: for example, according to this survey, consumers prefer a flatrate tariff (17% relative importance) over a one-time purchase (9% relative importance).

3.3 Cloud Service Quality Assurance: Qu4DS

To demonstrate the limitations of cloud service quality assurance from the perspective of provider SLA violation fee prevention, this section examines Qu4DS (Quality for Distributed Systems) as proposed by Freitas [FPP12].

3.3.1 Qu4DS

Qu4DS is an integrative approach to specifying and enforcing SLAs for cloud service providers, including "the creation of SLA templates [...], the

Attribute	Indicator	Relative Importance
Provider Reputation	Low reputation	0%
	High reputation	26%
Required Skills	No training required	7%
	Training required	0%
Migration Process	Use of standard data formats	21%
	Use of provider specific data	0%
Pricing Tariff	Pay-per-use tariff	0%
	Flatrate tariff	17%
	One-time purchase	9%
Cost compared to intern[al] solution	Equal costs	0%
	15% less costs	10%
	25% less costs	16%
Consumer Support	Individual electronic support	0%
	Standard electronic support	13%
	Individual personal support	11%

Table 4: Relative importance level of consumer preference attributes [KMA10, 7].

design of performance and fault-tolerance QoS assurance mechanisms [and] the translation of QoS to appropriate configurations of those mechanisms" [FPP12, 376]. Its goal is "to provide an autonomous service execution management while aiming at increasing the provider profit" [Fre12, 39]. Provider profit is increased by minimizing SLA violation fines and reducing the costs of infrastructure usage [FPP11, 117]. Qu4DS uses resources from an Infrastructure-as-a-Service provider to support the Software-as-a-Service layer and is therefor positioned in the Platform-as-a-Service layer [Fre12, 63], [FPP11, 117].

Qu4DS has been designed with a two-fold purpose:

- To provide autonomous service execution management;
- To increase provider profit by
  - Minimizing SLA violation fines; and
  - Reducing the costs of infrastructure usage.

In Qu4DS' approach, autonomous service execution management is realized by “self-adaptation mechanisms based on strategies that [react] to certain events at runtime [...] [i]n order to deal with the environment dynamism” [FPP11, 117]. SLA violation fines are minimized by selecting the service request most suitable for abortion due to lack of resources [FPP11, 117] and prioritizing more profitable consumers [Fre12, 35], while infrastructure usage costs are reduced by sharing the pool of booked resources among distinct contracts [FPP11, 117].

3.3.2 Qu4DS: Process

The following is a description of Qu4DS’ operation process [FPP11], depicted in Figure 5.

1. The process is initiated by a consumer contacting the Web Service (SaaS layer) employing Qu4DS in order to establish a Service Contract.
2. The resulting Contract Proposal is forwarded to the *SLA Negotiator* (PaaS layer), which
    (a) asks the *QoS Translator* (PaaS layer) to translate QoS to resource configuration which fulfills the Contract Proposal, and
    (b) checks through *Infrastructure Management* (between PaaS and IaaS layers) whether Resource Requirements can be met.
    (c) If the Resource Requirements can be met, the *SLA Negotiator*
        i. configures and deploys a service instance on the infrastructure through the *Job Management* interface (PaaS layer), and
        ii. commits contract agreement to the right consumers, which are now able to send Service Requests.
3. When a consumer sends a Service Request through the Web Service (SaaS layer), the Service Request is forwarded to the *SLA Negotiator*.
4. The *SLA Negotiator* then asks the
    (a) *Request Arrivals control loop** whether the Service Request can be treated;
    (b) If treatable, the Service Request is forwarded to the right service instance deployed on IaaS layer, which then
        i. prepares the distributed tasks necessary to treat request, based on configuration; and
        ii. asks Qu4DS to execute these tasks;
            - Qu4DS deploys the necessary tasks on the infrastructure through *Job Management* interface;

- The *Job Management* interface sends each task to the *Job Faults control loop** and the *Job Delays control loop**.

iii. If the necessary tasks are executed successfully, Qu4DS answers the service instance with the task results;

iv. the service instance finishes request treatment using tasks results and informs Qu4DS that the Service Request is treated.

v. the service instance forwards the Service Request result to the right consumer.

(c) *If any of the *Request Arrivals*, *Job Faults* or *Job Delays* control loops fail, the *SLA Negotiator* aborts the Service Request, informs the consumer about the SLA violation and computes violation penalties as agreed in the SLA.

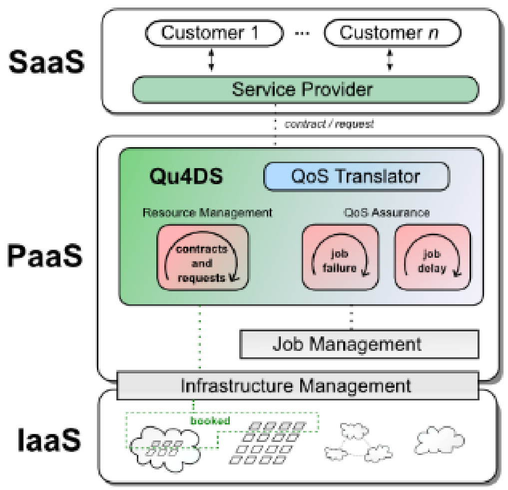

Figure 5: Qu4DS Process / Architecture [Fre12, 64].

3.3.3 Qu4DS: SLA Templates

In Step 1 of the Qu4DS process described in Section 3.3.2 a Service Proposal is negotiated between service consumer and provider. Qu4DS assumes this process to utilize SLA (contract) templates, offered by the provider and selected and customized by consumers [Fre12, 40]. Freitas offers contract templates based on two Quality of Service metrics measuring Performance and Fault Tolerance, respectively [Fre12, 41]:

- **Response Time**: The maximum amount of time allowable for request treatment; and

- **Reliability**: The degree of dependability.

The consumer's desired values for these metrics are expressed on a high level as *strong*, *medium* or *weak*; combining possible values for each metrics into four SLA templates, as shown in Table 5:

Template Label	Response Time	Reliability
Fast	*strong*	*weak*
Safe	*weak*	*strong*
Classic	*medium*	*medium*
Standard	*weak*	*weak*

Table 5: SLA contract templates [Fre12, 41].

3.3.4 Qu4DS: Performance Assurance

Qu4DS ensures Performance, as measured by the response time metric, by "deploying the [service] instance based on the minimal resource requirements able to meet the given response time" [FPP12, 379] through the following process [FPP12, 379]:

1. Following a Contract Proposal by the consumer, the service provider translates the required response time to its respective resource requirements based on profiling data;
2. The provider acquires resources from the infrastructure according to these resource requirements until the contract ends;
3. The translated resource requirements are used to configure and deploy the service instance;
4. The consumer can now send requests to service instance, which has been configured to the performance requirements.

3.3.5 Qu4DS: Fault Tolerance Assurance

Fault Tolerance is handled within Qu4DS by means of the *Job failures* and *Job delay* control loops. The goal of the Qu4DS fault tolerance assurance mechanism is "to improve the provider ability of overcoming malfunctions during request treatment" [FPP12, 397]. This is done by replacing failed and delayed jobs.

Job Failure Control Loop

Job failure occurs when the job process encounters a crash fault, for instance due to a non-successful I/O operation [Fre12, 45]. The Job Failure assurance algorithm employed in Qu4DS replaces failed jobs with a replacement job given that the *failure threshold* and *adaptation threshold* are not exceeded. The adaptation threshold is calculated by subtracting the execution time of a job from its request response time [Fre12, 47].

Job Delay Control Loop

Job delay occurs when the job's elapsed time exceeds its expected execution time. In Qu4DS Freitas assumes all delayed jobs to be failed jobs and discards the possibility of delayed jobs finishing eventually by using time constraints on service request treatment [Fre12, 46].

3.3.6 Qu4DS: Request Arrivals Control Loop

Freitas' Request Arrivals control loop checks if resources are available to treat a request; "If the resource reliability fits the request resource requirements, the request is treated. Otherwise, the request is aborted implying an SLA violation" [Fre12, 50]. Because it is Qu4DS' purpose to minimize provider cost [Fre12, 39], in case an ongoing request is using the same resources a request that has just arrived could use, the Request Arrivals loop "chooses which request will be aborted based on request fine abortion values, [...] aiming at minimizing the payment of fines" [Fre12, 50-51].

3.3.7 Qu4DS: Conclusion

Even though Qu4DS has the capability to set consumer-specific job failure and delay thresholds, Freitas applies instance configuration during service initiation, not during service consumption, as he assumes QoS requirements to stay the same during service provisioning [Fre12, 45]. As a result, Qu4DS' quality assurance process is not as dynamic as cloud services can be, as service instances employing Qu4DS have to be configured statically.

3.4 Summary

Cloud Computing offers software in a service-based manner as Infrastructure, Platform or Software to be consumed on demand. Standard network protocols are used to offer services to consumers on demand, scaling resources as needed, often on a pay-as-you-go basis. A service's life is cyclic in nature, allowing for improvements or service reconfiguration during service consumption. Service quality can be measured using quality attributes based on concepts such as performance, flexibility or reliability. The Qu4DS quality assurance method, attempting to minimize provider SLA violation prevention, is limited to assuring quality of preconfigured consumer requirements and is therefor incapable of assuring quality under conditions of dynamic requirements.

Before exploring the role quality attributes may play in consumer satisfaction of cloud service quality, Chapter 4 will explore existing Consumer Satisfaction theory.

Chapter 4

Theoretical Context: Consumer Satisfaction

In this chapter, existing Consumer Satisfaction theory is explored in order to form a basis on which to develop a methodology for assuring consumer satisfaction of cloud service quality. The nature of consumer satisfaction is explored, as well as the roles of consumer expectations and (service) quality or performance. Subsequently, several satisfaction metrics are discussed.

4.1 Formal Definition

In his book "Satisfaction: a Behavioral Perspective on the Consumer," Oliver has offered the following formal definition of the concept of satisfaction: "Satisfaction is the consumer's fulfillment response. It is a judgment that a product or service feature, or the product or service itself, provided (or is providing) a pleasurable level of consumption-related fulfillment, including levels of under- or overfulfillment" [Oli10, 13]. He notes that satisfaction can occur when a situation returns to normalcy or neutrality, as this invokes pleasurable fulfillment of the consumer's expectations. Important concepts incorporated in this definition are a consumer's fulfillment response and a judgment over a product or service (feature)'s *fulfillment*. A fulfillment involves at least two components: an *outcome* and a *referent* to compare this outcome to [Oli10, 14]. A consumer's response to their judgment of the level of fulfillment determines the level of satisfaction of a product or service (feature). Even if a consumer judges the level of fulfillment to be adequate, Oliver argues that a consumer can still be unsatisfied, since their *response* to the fulfillment can be one of unpleasantness: "many individuals find taxation dissatisfying [since] the fulfillment of this obligation is unpleasant" [Oli10, 14].

For purposes of this thesis, satisfaction will only be regarded in respect to services, not products.

4.2 Concepts

A potential consumer exploring the market with the intent of purchasing a certain service will aggregate information on the likely *performance* of a service, at the same time developing *expectations* of this performance. A consumer's choice of purchasing a specific service over others is based on a set of *choice criteria*. However, after the service has been purchased satisfaction no longer hinges on choice criteria but rather on *satisfaction drivers* [Oli10, 14]. Since choice criteria do not affect consumer satisfaction after the initial purchase, they fall outside the scope of this thesis.

After the initial service purchase, the consumer is in a position to compare the service's actual performance to their expectations and needs, resulting in an *expectation-performance discrepancy*, describing the difference between consumer expectation and service performance. In a satisfaction context, this discrepancy is referred to as *disconfirmation*. Similarly, a consumer can compare *perceived quality* to the actual *cost* of the service, resulting in a judgment of service *value* [Oli10, 19].

4.3 Comparison Operators

As discussed above, satisfaction is the outcome of a comparison between a service's performance and the consumer's response to the judgment of this performance. In regards to satisfaction, this judgment can be made based on several comparison operators. Each of these compares performance to a different aspect needing fulfillment that might influence consumer satisfaction. Oliver identifies the following six comparison operators [Oli10, 24]:

- Expectations;
- Needs;
- Excellence (Ideals);
- Fairness;
- Events That Might Have Been; and
- Nothing.

Expectations, as per Oliver's definition, are "prediction[s], sometimes stated as a probability [...], of attribute or product performance at a specific performance level" [Oli10, 28], in which performance level is defined as "[t]he perceived amount of product or service attribute outcomes received, usually reported on an objective scale bounded by good and bad levels of performance" [Oli10, 28]. This implies that, for the same service, a consumer can have a different set of expectations of a certain attribute's performance at the performance level of "abundant information provision towards the consumer" than at the performance level of "poor information provision towards the consumer."

At the most elementary level, *Needs* can be defined as "requirement[s] to fulfill a goal" [Oli10, 28]. The reason behind a service being consumed is essentially because it is expected to fulfill the consumer's goals. There can be two fundamentally different purposes for service consumption; either *restoration* or *enhancement*. The former restores the consumer's perceived state to the minimum level of wholeness, while the latter adds to the positive value of the consumer's perceived state, which already includes all the necessary essentials. This distinction aligns with the distinction between needs and wants; needs describe the requirements to fulfill restoration, while wants describe the requirements to fulfill enhancement [Oli10, 136]. Consumer satisfaction in terms of Needs is achieved through *needs fulfillment*.

The *Excellence (Ideals)* operator approaches satisfaction through the judgment of quality, or more specifically as "a comparison [of the quality judgment] to the consumer's excellence standards" [Oli10, 28]. The consumer's excellence standards signify the level at which they perceive the service to be of very high quality, while the consumer's ideal standards signify the level of performance expected under ideal / unrestricted circumstances (for example without cost restrictions).

The comparison operator *Fairness* approaches satisfaction from a justice point of view. A consumer may be unsatisfied with a provided service if they feel other entities gain more rewards for investments in the service similar to those done by the consumer. A key term in Fairness is (in)equity; a "fairness, rightness, or deservingness comparison to other entities, whether real or imaginary, individual or collective, person or non-person" [Oli10, 194]. Basically, a consumer may be satisfied with a service in terms of fairness even if the returns for a high investment were nearly zero, as long as other parties gained similar rewards. However, if one party gained high rewards from the same investment, other consumers will be dissatisfied with the service (or its provider) due to a feeling of injustice being done towards them. Even though the Fairness comparison operator influences consumer satisfaction with a service, consumer dissatisfaction with a service for Fairness reasons is not a result of unsatisfactory service performance or quality, but rather unsatisfactory treatment of the consumer by the service provider compared to other consumer, in spite of the service's performance or quality.

Oliver's comparison operator *Events That Might Have Been* considers buyer's remorse or *regret*, which is a comparison between "what might have been" and "what is," or "a comparison to alternative outcomes that could have been likely or could have been foreseen" [Oli10, 217]. The consumer compares the actual results of service purchase to their predicted results of purchasing this service, an alternative service or no service at all. This might result in the conclusion that the actual outcome is not as satisfactory to the consumer as their predicted outcome would suggest. In case the actual outcome of the purchase is not as satisfactory as the predicted outcome of alternative or no service purchase, the consumer experiences *regret* for realizing "something else would have been better than what I selected." In case the actual outcome of the purchase is not as satisfactory as the predicted outcome of this very purchase, the consumer can experience *hindsight* after reconstructing their predictive expectations and realizing their predicted outcome was unrealistic; the consumer could have known their choice would lead to this unsatisfactory outcome, and "should have behaved differently and avoided it" [Oli10, 216-217]. Both regret and hindsight result in dissatisfaction with the consumed service, but this dissatisfaction is a result of the consumer misinterpreting their expectations rather than disappointing service performance or quality.

The last comparison operator, *Nothing*, "acknowledges the possibility that [service] performance can affect satisfaction directly if no comparison operators are considered" [Oli10, 24].

For the purposes of this thesis, only the first three of these comparison operators (*Expectations*, *Needs* and *Excellence (Ideals)*) are interesting, as *Fairness* and *Events That Might Have Been* do not necessarily deal with satisfaction in respect of service quality, but primarily incorporate consumers' personal morals and emotions into service satisfaction. The last operator, *Nothing*, corresponds with the point of view that "an increase in service performance equals an increase in consumer satisfaction" and is therefor not influenced by the consumer; consequently, this way of achieving satisfaction falls outside the scope of this thesis as well. Furthermore, it can be argued that *Expectations* and *Needs* are complementary to one another, since a consumer might initially choose to purchase a service that can be expected to fulfill their needs; if a service's performance fails to meet the consumer's needs, it automatically fails to meet the consumer's expectations as well: "[o]ften [the] expectation and need will overlap exactly, becoming interchangeable" [Oli10, 68]. This assumption will be made in order to further limit the scope of the satisfaction section of this thesis. Accordingly, this thesis will define consumer satisfaction solely based on the comparison between service performance and the *Expectations* and *Excellence* operators, as illustrated in Figure 6.

The role of *Expectation* and *Excellence* in a consumer's (dis)satisfaction with a service is examined in the theory of the Expectancy Disconfirmation Paradigm, currently the dominant theoretical paradigm in consumer satisfaction [Oli10, 23], as discussed in Section 4.4.3.

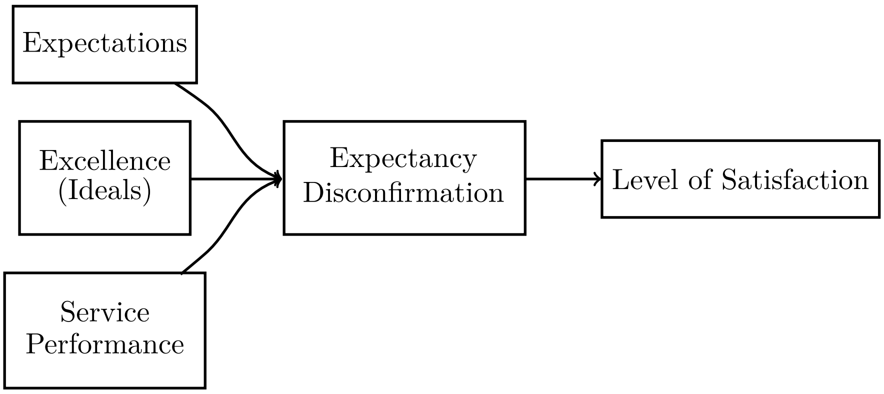

Figure 6: Satisfaction Model based on Expectations and Excellence [Oli10].

4.4 The Role of Expectations in Consumer Satisfaction

4.4.1 What is an expectation?

As previously noted, Expectations, as per Oliver's definition, are "prediction[s], sometimes stated as a probability [...], of attribute or product performance at a specific performance level," in which performance level is defined as "[t]he perceived amount of product or service attribute outcomes received, usually reported on an objective scale bounded by good and bad levels of performance" [Oli10, 28]. The role expectations play in service consumption is twofold; expectations play a guiding role in the selection of services to be purchased: the service expected to fulfill the consumer's needs to the greatest extent would be the service selected for purchase. After the initial service purchase, the consumer's satisfaction with the service is based on the comparison between the initial expectations of that service and the service's quality or performance. This definition of expectations is very broad. For example, at different performance levels (ranging from excellent through adequate to bad in any service attribute) different expectations of a service attribute may exist, while these expectations are not all of interest with respect to consumer satisfaction. Additionally, the way in which expectations are compared to service performance may influence the diagnosis of a consumer's satisfaction level. Therefor, a more elaborate definition of Expectations is needed.

4.4.2 Classification of Expectations

If satisfaction is defined as a comparison between actual service outcome and a consumer's initial expectations thereof, a clear point of reference needs to be defined against which the actual service outcome can be compared to. The distance between this point of reference and the outcome of the respective service performance attribute can then be used to quantify the level of satisfaction at this respective reference level. In terms of expectations, such a reference point is called an *expectation referent*. Oliver [Oli10] has distinguished three expectation referent categorizations, which will be elaborated on in the following sections:

1. Expectation referents categorized by *level of desire*;
2. Expectation referents categorized by *level of abstraction*; and
3. Expectation referents categorized by *focal comparison objects*.

Expectation referents categorized by level of desire

According to Oliver, Miller offered the opportunity to match the initial expectation of service performance to its outcome by proposing to categorize expectation referents by level of desire [Oli10, 70][Mil77]:

1. Ideal or wished-for level ("can be");
2. Expected or predicted level ("will be");
3. Minimum tolerable or lowest acceptable level ("must be");
4. Deserved level ("ought to be").

In monopoly situations, it is possible for the expected or predicted level to fall below the minimum tolerable level, but generally it will fall between the ideal and minimum tolerable levels. The deserved level of expectancy is based on what consumers feel is appropriate to be expected, based on their investments, rights and position. The range between minimum tolerable and ideal levels is referred to as the *zone of tolerance* [ZBP91][Oli10, 70], as depicted in Figure 7. In case the performance of a service falls below the consumer's zone of tolerance, dissatisfaction with the purchased service is as good as guaranteed. However, this does not necessarily entail discontinuation of service consumption, as consumers may not have (direct) access to alternatives [Oli10, 70].

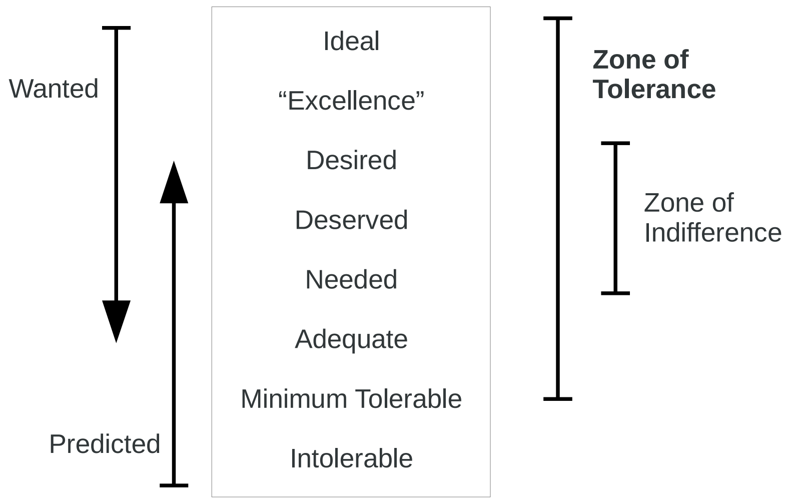

Figure 7: Zone of Tolerance in Desirability Levels [Oli10, 72].

Expectation Referents Categorized by Level of Abstraction

Expectations do not necessarily have to be concrete; the consumer can be unaware they have a certain expectation until it remains unsatisfied, or they can be ambiguous about certain expectations [OW87]. Oliver and Winer [OW87] have presented a conceptual discussion of expectations based on Passive versus Active Expectations, Knowable versus Unknowable Outcomes and Levels of Certainty regarding expectations.

These conceptual expectations are divided into three tiers of distinction. The first layer is that of *Passive* versus *Active Expectations*. Passive expectations are those expectations that, even though a consumer might be aware of the existence of certain possible outcomes, are not processed by the consumer as outcomes probable to be encountered by them. Oliver offers the example of consumers not acknowledging the possibility of a refrigerator maintaining warm temperatures in their expectations of said refrigerator's function, while at the same time being fully aware that in case of a malfunction the outcome will be exactly that; this unprocessed outcome is known as a passive expectation [Oli10, 73].

Active expectations can be further divided into expected *Knowable Outcomes* and considered *Unknowable Outcomes*. Oliver [Oli10, 74-75] argues that in cases of a product or service being experimental or innovative, the consumer can expect to encounter unknowable outcomes; they realize there is no means to know what possible outcomes could arise, since no preexisting knowledge is available. The consumer will have to accept being *Ignorant* to these possible but unknowable outcomes. On the other hand, when preexisting knowledge *is* available, the consumer can classify their active expectations with knowable outcomes according to probabilities based on their own or other consumers' previous experiences. For example, if a consumer has deduced from previous experiences that gas prices at gas stations on the highway generally lie 10 to 15 cents higher than the prices at local gas stations, they can reasonably expect with *Certainty* that this will also be the case today. In other situations, consumers may expect knowable outcomes based on previous experiences or data with *Probability* (for example the chance of winning the lottery), or with *Ambiguity*, in which case "the possible outcomes of a purchase are known, [but] the probabilities of their occurrence are not" [Oli10, 74].

For the purposes of this thesis, expectation referents categorized by level of abstraction are not relevant, since consumers rely on these referents solely *before* purchasing a service in order to assess their expectations of the outcome of their *purchase* of a service, rather than their expectations of the outcome of the *performance* of a purchased service.

Expectation Referents Categorized by Focal Comparison Object

In addition to comparing service attribute performance directly to a consumer's expectations, it can also be compared to another object, such as another service or service attribute, where the original service attribute is "expected to outperform or meet the performance of an alternative or comparative referent *object*" [Oli10, 75]. This can be done in several ways: performance can be compared to *other brands* in the same product class or to a product (class) norm; new performance can be compared to *previous performance*; consumer states can be compared to *states of other consumers*; performance can be compared to performance of the *same attribute in different situations*; and performance may be compared to consumer's *internal standards* or to *external claims* (for example service providers' claims of service (attribute) performance) [Oli10, 75-76]. These are all examples of service (attribute) performance expectations compared to Focal Comparison Objects.

4.4.3 The Expectancy Disconfirmation Paradigm

Oliver's Expectancy Disconfirmation Paradigm [Oli10, Ch. 4] is based on the comparison between consumer expectations of service performance and the actual performance of this service or attributes thereof. The difference between expectation and performance is known as *disconfirmation*, which can either be positive (in case performance is higher than expected), or negative (in case performance is lower than expected). When performance is equal to the consumer's initial expectations, a *confirmation* of expectations exists [Oli10, 104]. Situations leading to the different types of disconfirmation are represented in Table 6.

Disconfirmation	**Consumer's Expectations**
Positive	Low-probability desirable events occur and/or high-probability undesirable events do not occur.
Zero	Low- and high-probability events do or do not occur, as expected.
Negative	High-probability desirable events do not occur and/or low-probability undesirable events occur.

Table 6: Categories of Disconfirmation and States of Nature [Oli10, 104].

Oliver's Expectancy Disconfirmation Model can be mathematically represented as follows [Oli10]:

$$\text{Disconfirmation} = P_i - E_i \tag{4.1}$$

where $P_i =$ the actual performance outcome of attribute i; and
$E_i =$ the performance outcome of attribute i as expected by the consumer.

Subjectivity of Expectancy Disconfirmation

Expectancy disconfirmation is a subjective measure, rooted on the expectations of the consumer involved. Two separate consumers may have different initial expectations of the same performance attribute, while the actual performance attribute's outcome will be the same for both consumers. In the event of the actual performance attribute outcome falling in between the expectations of the two consumers, this would result in positive expectation disconfirmation for one consumer, while resulting in negative disconfirmation for the other [Oli10, 106-107].

However, this does not necessarily result in the first consumer being satisfied with the performance attribute's outcome and the second consumer being dissatisfied; it merely implies performance is better or worse *than expected*. The first consumer's positive disconfirmation might stem from them having very negative initial expectations of the attribute's outcome, which have been disconfirmed by a slightly better actual performance outcome. Nevertheless, this actual outcome might still fall below the lowest acceptable level of performance (see Figure 7) this consumer considers satisfactory. Oliver illustrates this phenomenon in Figure 8. The same principle applies for very positive initial expectations: negative disconfirmation of high positive initial expectations does not necessarily result in dissatisfaction.

(see Figure 9). In such cases, relatively high levels of disconfirmation are necessary to overcome the initial level of optimism or pessimism regarding performance in order for positive and negative expectancy disconfirmation to result in satisfaction and dissatisfaction, respectively. In cases of more moderate initial performance expectations, smaller disconfirmation levels can cause shifts from positive expectations to dissatisfaction or negative expectations to satisfaction. Henceforth, as expectations deviate less from neutrality, expectancy disconfirmation, and therefor actual performance (attribute) outcome, have an increasing effect on consumer (dis)satisfaction.

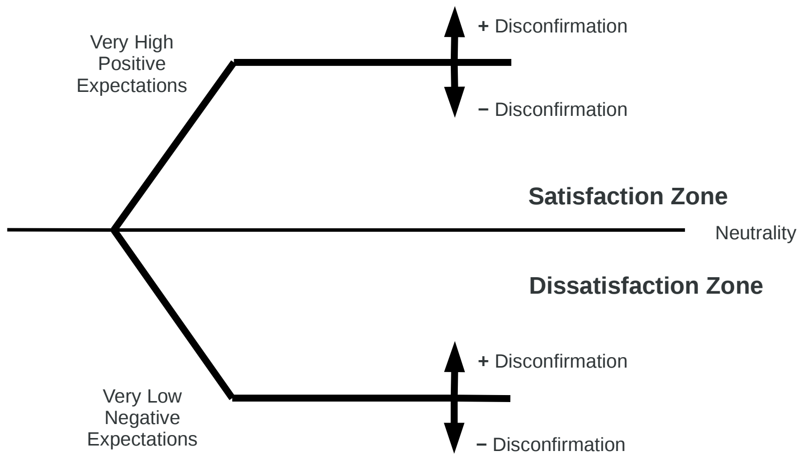

Figure 8: Disconfirmation under strong expectations and weak disconfirmation [Oli10, 115].

4.5 The Role of Quality in Consumer Satisfaction

4.5.1 Technical Quality

Traditionally, companies strive to offer product and service quality by producing precisely according to technical specifications; an approach known as *conformance quality* [Oli10, 163]. A negative aspect of this approach is the lack of consideration for the consumer's perspective of quality; products or services of high technological quality are assumed to satisfy the consumer by definition, regardless of the consumer's needs, expectations or preferences. Although conformance quality to an extent corresponds with consumer preferences, it does not necessarily result in consumer satisfaction, as consumers might be conscious of potential alternatives of even higher quality [Oli10, 163].

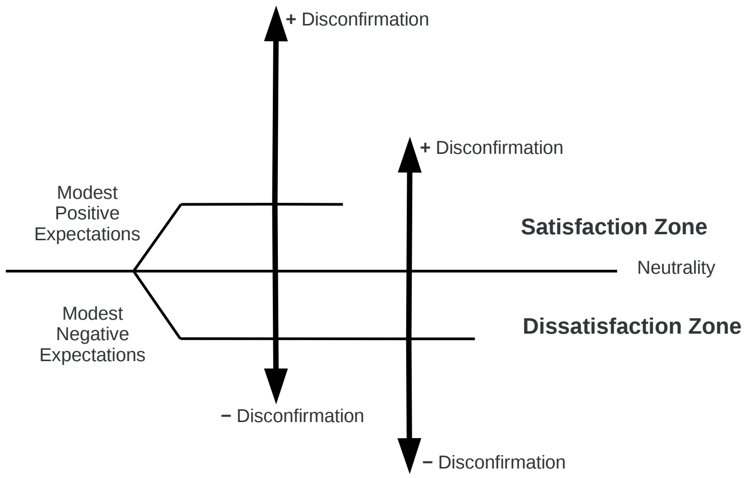

Figure 9: Disconfirmation under weak expectations and strong disconfirmation [Oli10, 117].

4.5.2 Quality as Perceived by the Consumer

Oliver recognizes a distinction between single-stimulus and dual-stimulus definitions of quality [Oli10, 165-167]. Single-stimulus definitions use a single term to define an aspect of quality, resulting in an immeasurable subjective referent for quality. For example, if the quality of a design or blueprint is defined as *detailed*, ambiguity may still emerge over the *level* of detail; how finely detailed is suggested by the definition of 'detailed' [Oli10, 165]? Moreover, seemingly singular quality definitions, such as *superior* imply the comparison of quality to an external referent and are therefor not legitimately singular. Consequently, single-stimulus definitions are inappropriate for the use of objectively measuring quality as perceived by the consumer as they contain implied and therefor untraceable referents.

Dual-stimulus definitions, on the other hand, incorporate the nature of quality definition as being comparative by specifying a *comparative referent* in the definition of quality. This is achieved by comparing the *performance dimension* of quality to a relative *standard* of quality, corresponding to the underlying needs of the consumer. For example, a consumer's need to keep costs to a minimum, or a reference to the consumer's Ideal quality levels (see Figure 7), enables an objective measurement of quality definition, even if this definition contains subjective dimensions. The quality of the design or blueprint (see previous paragraph) can now be defined as *as detailed as the budget allows for*, which is objectively measurable in case the size of the budget is known.

4.5.3 Measuring Quality

Comparing Actual Performance to Expectations

Parasuraman et al. [PZB88] have developed a formula for measuring quality in order to determine a best-brand company in a service category. This formula compares a company's service actual performance against "an imagined or real company possessing essential and excellent (not ideal) levels of features" [Oli10, 170]. This formula compares a company's service *actual* performance to how it *should* perform. Parasuraman et al. have defined this difference as a "quality gap" in the SERVQUAL instrument [PZB88, 19][Oli10, 170]:

$$Q_j = \sum (P_{ij} - E_i) \tag{4.2}$$

where Q_j = the quality gap for company j;
$\sum$ = a summation over all dimensions, features, or attributes;
P_{ij} = the actual performance perception for company j on dimension or attribute i; and
E_i = the excellence expectation for dimension or attribute i.

Comparing Actual Performance to Ideal Point

Oliver has set up a formula in which a company's service quality is measured by comparing the actual performance of that company's service attributes to the performance of the consumer's ideal company's service attributes. Oliver assumes over-performance (better than ideal) of an attribute to contribute to quality negatively, accounting this to the phenomenon of "too much of a good thing" [Oli10, 169]. Oliver explains this phenomenon using the performance of a refrigerator: if it over-performs by cooling too well, the products in the fridge are likely to freeze, an undesirable performance outcome [Oli10, 169] . Oliver's formula comparing actual performance to ideal performance [Oli10, 169] is defined as follows:

$$Q_j = 100 - \sum |P_{ij} - I_i| \tag{4.3}$$

where Q_j = the quality judgment of company j;
$\sum$ = a summation over all attributes;
P_{ij} = company j's actual performance on attribute i; and
I_i = the consumer's ideal company's performance on attribute i.

Note: Since this formula designates a "perfect" level of quality as being 100, P_{ij} and I_i should be used in the same scale of [0-100].

The "Modified Quality" Model

Ideally, in the absence of restrictions such as costs, a consumer would prefer an *ideal* product, a product "which possesses ideal levels of all its relevant features" [Oli10, 168]. Assuming this, the overall quality of a product or service can be measured by collectively comparing the actual performance of all relevant features to the ideal levels of these features. Teas [Tea93] has proposed a formula describing the measurement of quality based on a comparison between actual performance, expected performance and ideal performance levels of quality attributes called the *Modified Quality* formula [Tea93, 20]:

$$MQ_i = -(|P_i - I| - |E_i - I|) \qquad (4.4)$$

where MQ_i = the Modified Quality measurement attribute i;
P_i = the actual performance of attribute i;
I = the ideal level of an attribute; and
E_i = the expected performance level of attribute i.

Note: Recognizing that Teas argues the use of I instead of I_i since "an attribute's ideal level is constant" [Tea93, 20], for purposes of distinction between values of I for different attributes, this thesis will define I_i as "the constant value of a consumer's ideal level of performance of attribute i" and uses I_i interchangeable to Teas' I.

In case actual performance is equal to expected performance ($P_i = E_i$), $MQ_i = 0$. In case actual performance is better than expected performance but both are worse than the Ideal level ($I_i > P_i > E_i$), $MQ_i > 0$. In case actual performance is worse than expected performance and both are worse than the ideal level of performance ($I_i > E_i > P_i$), $MQ_i < 0$.

Short-term and Long-term Quality Measurement

Quality can be measured at two consumption stages: overall / global quality judgments and short-term quality judgments, such as during transactions or at the encounter of a certain event. Short-term quality judgments can be measured using a simple summation over performance ratings $\sum P_{ij}$, where P_{ij} is the actual performance of attribute i of service j, as defined in either Oliver's ideal-point formula (4.3), SERVQUAL's expectations of excellence formula (4.2), or Teas' Modified Quality Formula (4.4). When some experience has been gained on probable actual performance outcomes of certain attributes, formulas (4.2) and (4.3) can be adapted to reflect these probabilities by adding a probability coefficient pr_{ij} [Oli10, 174-175]:

$$Q_j = \sum pr_{ij}(P_{ij} - E_i) \qquad (4.5)$$

and

$$Q_j = 100 - \sum pr_{ij}|P_{ij} - I_i| \qquad (4.6)$$

where pr_{ij} = the probability of attribute i of service j performing at the P_{ij} level.

Since measurement of quality on a global level is typically dimension-free, Oliver argues [Oli10, 175-176], quality attributes are left unconsidered so that any one attribute cannot influence a consumer's judgment disproportionally. Therefor, "as long as attributes or dimensions are present in the measure, whether it is one- (performance-only) or two-dimensional, global measurement will be compromised by implicit weighting of dimensions by individual consumers [while] proper measurement of quality permits investigation of its [dependence] with other concepts"[Oli10, 176]. Oliver points out that since in academics "little attention [has been] paid to the measurement of overall quality beyond the simple summation of attribute ratings" [Oli10, 176], additional research on this aspect is necessary. Using weighted dimensions of quality, the overall quality of a service as perceived by the consumer can be calculated by summing up the Modified Quality measures of each service attribute. If the consumer has specified certain priorities in the Service Requirements Specification phase, these can be accounted for by introducing weights w_i to the attributes' Modified Quality measures [Oli10, 168]:

$$MQ = \sum(\frac{w_i}{w} MQ_i) \tag{4.7}$$

4.5.4 Quality and Satisfaction

Although the concepts are closely affiliated, Oliver identifies several conceptual differences that distinguish quality and satisfaction. The concept of *Experience Dependency* distinguishes between quality and satisfaction [Oli10, 177-178]: satisfaction is purely dependent on experience, as the level of need fulfillment can only be measured if a sense of fulfillment has been experienced. Quality, on the other hand, can be *perceived* through perceptions such as the satisfaction of others with a product service, or through descriptions of products or services.

Another distinction between quality and satisfaction lies in the nature of quality and satisfaction attributes and dimensions: "For any given product or service, there will be some degree of consensus as to what the relevant quality dimensions are. [...] Satisfaction judgments, in contrast, can result from any dimension, quality-related or not" [Oli10, 178] (see Figure 10).

A similar distinction lies in expectations of quality and satisfaction attributes: "[T]he standards used for quality judgments are based on ideals or excellence perceptions, [while standards used in satisfaction judgment include] predictive expectations, needs, product category norms, and even expectations of quality" [Oli10, 178] are not necessarily based on ideal or excellent levels.

On a more basic level, satisfaction is influenced by emotion as well: "Quality judgments, being largely attribute-based, are thought to be primarily cognitive[...] [, while] satisfaction is [...] thought to be both a cognitive and an affective response" [Oli10, 178].

Although there are distinct differences between quality and satisfaction, the two concepts are strongly related. Encounter-specific and global judgments both have elements of quality and satisfaction judgments. At the encounter-specific level, quality judgments result from a comparison of actual performance to ideal or excellence standards. These encounter-specific judgments of quality have an influence on the encounter-specific satisfaction (as shown in 4.4.3) and, accumulated over multiple encounters, will provide an impression of global quality. Similarly, accumulated satisfactory encounters will influence the global quality perception. Global satisfaction is a strong influence on the perception of global quality, as satisfaction over a longer period of time is assumed to create the perception that a product or service providing such a level of satisfaction must be of high quality. Similarly, a high level of perceptions of quality encounters will likely lead to a sense of satisfaction on the global level [Oli10, 181-182].

4.5.5 An Encounter Quality-influences-Satisfaction Model

Aggregating the previous theories of consumer satisfaction and the role of quality therein, Oliver proposes "an encounter-specific quality and satisfaction model" [Oli10, 187], as presented in Figure 10. This model represents how the disconfirmation of expectations of Ideal levels of performance influence quality perception directly, rather than the satisfaction judgment, while disconfirmation of predictive (non-ideal) expectations of both quality and non-quality attributes influence the satisfaction judgment. Other comparison operators as well as the perception of quality itself may influence the consumer's judgment of satisfaction.

It is important to note that this model only encompasses service encounters or aggregates thereof; it does not reflect quality or satisfaction on a global level. Moreover, it does not account for judgments of quality or satisfaction stemming from an affective response; it is purely performance-driven. Furthermore, the possibility that situations occur in which consumers are satisfied with low quality or dissatisfied with high quality (see Figures 8, 9), even though not explicitly clear in this model, should be taken into account when determining levels of consumer satisfaction.

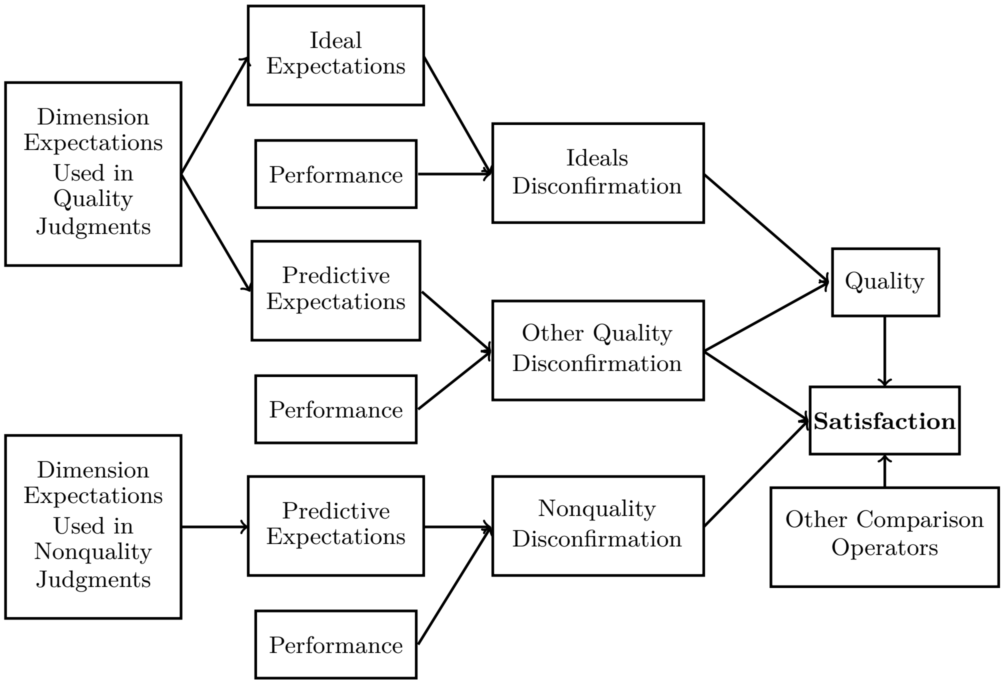

Figure 10: Oliver's Encounter-Specific Quality and Satisfaction model [Oli10, 187].

4.6 Summary

In this chapter, a definition of consumer satisfaction has been presented and its objective nature has been investigated. Comparison operators have been used to determine *what* a consumer might be (un)satisfied with. The role of consumer expectations in satisfaction and the different levels of expectations within and outside a consumer's Zone of Tolerance have been explored. The difference between a consumer's expectations of and the actual performance of a service has been defined as Expectation Disconfirmation, while it has been made clear that even if a service outperforms a consumer's expectations, it may still be qualified as "unsatisfactory" by the consumer. The long- and short-term influence of service quality as perceived by the consumer on consumer satisfaction has been distinguished. Finally, an overall model has been introduced which depicts the dimensions influencing a consumer's level of satisfaction with a service.

The theories discussed in this chapter and Chapter 3 form the basis for the development of the ACCQ-methodology in Chapter 5, addressing the assurance of consumer satisfaction of cloud service quality during service consumption.

Chapter 5

Methodology for Assuring Consumer Satisfaction of Quality during Service Consumption (ACCQ-Methodology)

Based on the theories described in the previous Theoretical Context chapters (2, 3, 4), this chapter presents the ACCQ-methodology (Assuring Consumer satisfaction of Cloud service Quality), which prescribes what is required of a service provider or assurance method in order to be able to assure consumer satisfaction of cloud service quality during service consumption.

Section 5.1 of this Chapter defines what it entails to assure satisfaction *during service consumption*. Section 5.2 defines the context of cloud service *quality*. Section 5.3 defines the first step of the ACCQ-methodology, describing how to determine which aspects of cloud service quality influence consumer satisfaction during service consumption. Section 5.4 defines the second step of the ACCQ-methodology, describing what is required of a service provider or assurance method in order to assure consumer satisfaction of cloud service quality during service consumption.

5.1 Cloud Service Lifecycle

Based on the cloud service lifecycle theories of Joshi et al. [JFY09] and ITILv3 [CHR+07], [Arr10] on a cloud service's lifetime cycle, this section presents a definition of a lifetime cycle from the perspective of the consumer, from here on referred to as *“Consumer Perspective Lifecycle.”*

5.1.1 Lifecycle Phases from the Consumer’s Perspective

The phases in ITILv3 and Joshi et al.’s service lifecycles slightly correspond and are therefor combined into the “Consumer Perspective Lifecycle”, as shown in Table 7.

ITILv3’s Service Strategy phase includes the Requirements Specification phase as defined by Joshi et al. as well as the tasks of developing policies, guidelines and processes to be used to manage the service(s) [Com08, 12]. As the latter tasks do not involve the consumer, they will be left out of the phase known in the Consumer Perspective Lifecycle as the *Service Requirements Specification* phase, in which the consumer defines the service domain and sets technical, functional and non-functional performance expectations, specifications and priorities, recorded in a Service Level Agreement with the service provider.

Joshi et al.’s Service Discovery and Negotiation phases fall within the scope of ITIL’s Service Design phase. The Service Composition phase as defined by Joshi et al. falls between ITIL’s Service Design and Transition phases; the bundling of services is part of the Service Design phase, while configuring the bundled services as one composed service falls within the Service Transition phase. As the Discovery and Negotiation phases as well as the bundling and configuring of combined services rarely involve end consumers and since the results of these steps are usually imperceptible to end consumers, the *Service Design* phase of the “Consumer Perspective Lifecycle” will be defined as service discovery, negotiation and composition as performed by the service provider, resulting in the consumer waiting for the service to be designed.

The Consumer Perspective Lifecycle phase *Service Transition* will be comprised of ITIL’s Service Transition phase minus the Service Composition factors and accordingly be defined as the phase in which the new or changed, possibly bundled service developed in the Service Design phase are transitioned into the Service Consumption phase. This may include transferring the consumer’s data from an old system or service to the service-to-be-consumed or configuring the new service to the end consumer’s specifications.

The *Service Consumption* phase of the “Consumer Perspective Lifecycle” corresponds with ITIL’s Service Operation and Joshi et al.’s Service Consumption phases and includes service delivery, quality monitoring and maintenance and compensation (payment).

In the Consumer Perspective Lifecycle, the *Continual Service Improvement* phase is adopted from the ITILv3 cloud service lifecycle. This phase represents the continuous measurement and improvement of service levels, technology and efficiency and effectiveness of processes of the service throughout all service lifecycle phases in order to meet the consumer's specifications and needs.

<table>
<tr><th>ITILv3 [CHR+07]</th><th>“Integrated Lifecycle”[JFY09]</th><th>“Consumer Perspective Lifecycle”</th></tr>
<tr><td>Service Strategy</td><td>Requirements Specification</td><td>Service Requirements Specification</td></tr>
<tr><td rowspan="2">Service Design</td><td>Service Discovery
Service Negotiation</td><td rowspan="3">Service Design</td></tr>
<tr><td rowspan="2">Service Composition</td></tr>
<tr><td rowspan="2">Service Transition</td></tr>
<tr><td></td><td>Service Transition</td></tr>
<tr><td>Service Operation</td><td>Service Consumption</td><td>Service Consumption</td></tr>
<tr><td>Continual Service Improvement</td><td></td><td>Continual Service Improvement</td></tr>
</table>

Table 7: Service Lifecycle Phases.

Consumer satisfaction assurance takes place during the Consumer Perspective Lifecycle phases Service Consumption and Continual Service Improvement. During the Service Requirements Specification and Service Design phases, the capabilities required of a service provider or assurance method as prescribed by the ACCQ-methodology (see Sections 5.3 and 5.4) need to be incorporated into the service in order to support consumer satisfaction assurance.

5.2 Service Quality Attributes

Based on the quality attributes accumulated in Section 3.2.2, Table 8 defines dimensions and corresponding attributes of cloud service quality. The following adaptations have been made to Table 2 to construct Table 8:

- Benlian et al.'s [BKH11] *Features* dimension (see Table 3) has been added as a whole;
- Attribute *Provider Reputation* from Table 4 has been added to the Rapport quality dimension;

- Attribute *Required Skills* from Table 4 has been added to the Features quality dimension;
- Attributes *Consumer support*, *Technology support availability*, *Budget control* and *Help functionalities* have been combined into quality attribute *"Consumer support"* under the Rapport dimension;
- Attributes *Training system* and *Required skills* have been combined into quality attribute *"Required skills"* under the Rapport dimension;
- Attributes *Data backup*, *Fault recovery strategy*, *Regular security audit* and *Anti-virus tool* have been combined into quality attribute *"Security measures"* under the Security dimension;
- Attribute *Data reporting and extraction features* from the Features dimension has been moved to the Flexibility dimension;
- Attribute *Configuration features* from the Features dimension has been moved to the Flexibility dimension;
- Attribute *Dashboard features with customer's service usage metrics* from the Features dimension has been split up; the *Dashboard features* portion has been moved to the *Configuration features* attribute and the *Customer's service usage metrics* portion has been moved to the *Budget control* attribute.

The remaining quality attributes found in Table 4 are either already included in Table 2 or represent service costs and pricing. Service cost and pricing is considered outside the scope of consumer perception of service quality for purposes of this thesis as they represent deliberate choices made prior to service consumption; consumer expectation or service performance of these attributes are not expected to change during service consumption.

Dimension	**Attribute**	**Source**
Rapport	Required skills	[HZ13]
	Customized service	[HZ13]
	Consumer support	[HZ13], [KMA10], [BKH11]
	Provider Reputation	[KMA10]
Responsiveness	Dynamic scalability response	[HZ13]
	Disaster recovery time	[HZ13]
Reliability	Elastic service availability	[HZ13]
	Service accuracy	[HZ13]
Flexibility	Multi-client access adjustment	[HZ13]
	Extra resources allocation	[HZ13]
	Data migration	[HZ13], [KMA10]
	Data reporting and extracting features	[BKH11]
	Configuration features	[BKH11]
Security	Security measures	[HZ13]
	Data secrecy	[HZ13]
	Access control	[HZ13]
Features	Visually appealing, sympathetic user interface	[BKH11]
	User-friendly navigation structure, search functionality	[BKH11]
	Core features supporting process steps / activities	[BKH11]

Table 8: Aggregated Service Quality Dimensions.

5.3 ACCQ-methodology: Step 1

In order to be able to assure consumer satisfaction of cloud service quality during service consumption, it is important to determine which of the quality attributes found in Table 8 can influence consumer satisfaction *during service consumption* and which of these attributes do *not* influence satisfaction during service consumption. Similarly to attribute Service cost and pricing method (see Section 5.2), consumer requirements of some of these attributes are in general not to be expected to change during service consumption, but only play a role during service design or negotiation. Due to the dynamic nature of services, requirements of other attributes can be very reasonably expected to change during service consumption. In order to distinguish between these two types of attributes, they will be classified as "design choice" and "dynamic" quality attributes, respectively. Since "design choice" attributes do not influence consumer satisfaction *during service consumption* but only play a role *prior to* service consumption (see "choice criteria" in Section 4.2), satisfaction assurance methods need to be capable of assuring satisfaction of "dynamic" attributes only. The classification of attributes in this manner can be done using the following consideration: *"Can the consumer requirements of attribute i reasonably be expected to change during service consumption without the nature of the service inherently changing?"* In this thesis, this classification is done *in the general case*. In case this methodology is applied for satisfaction assurance of *a specific service*, this classification needs to be performed respective to the specifications of this specific service. For example, in general the consumer requirements of quality attribute Configuration features cannot reasonably be expected to change during service consumption, as changing a barely configurable service to a highly configurable service inherently changes the nature of the service, necessitating a service redesign. For this reason, attribute Configuration features is classified as a "design choice" attribute in this thesis, as this is generally the case. However, if a service consumer has specified the service needs to be capable of dynamic Configuration features adaptation, changes in Configuration features requirements *can* reasonably be expected to change during service consumption, in which case this attribute needs to be classified as "dynamic" and consumer satisfaction of this attribute needs to be assured.

5.3.1 "Design choice" Quality Attributes

The following quality indicators found in Table 8 function as indicators of design choice more than they do as indicators of consumer requirements. They distinguish specific design choices; the requirements and/or performance of these indicators are not expected to change significantly during service consumption. For this reason, the following indicators will not be considered as indicators of consumer perception of service quality during service consumption for the purposes of this thesis:

- Required skills;
- Level of customized service;
- Provider reputation;
- Configuration features;
- Elastic service availability;
- Multi-client access adjustment;
- Extra resources allocation;
- Security measures; and
- Visually appealing, sympathetic user interface.

5.3.2 "Dynamic" Quality Attributes

The quality attributes shown in Table 9 may have an influence on consumer satisfaction of cloud services during service consumption. Changes in consumer requirements of these attributes can reasonably be expected to occur during service consumption, resulting in the service serving the same function but under different constraints. Furthermore, service environment fluctuations may have an influence on a consumer's perception of satisfaction regarding these quality attributes.

5.4 ACCQ-methodology: Step 2

In order to assure consumer satisfaction of cloud service quality during service consumption, consumers' satisfaction of each "dynamic" quality attribute, as found in Table 9, needs to be assured. Based on the theories discussed in Chapters 3 and 4, this can be done using the following method:

1. Define a desired (lower) limit for consumer satisfaction of service quality against which the assurance method will be assessed (for example "consumer classification of service quality as at least *satisfactory*").
2. For each quality attribute i:

Dimension	**Attribute**	**Indicator / Metric**
Rapport	Consumer Support	SLA performance
Responsiveness	Dynamic scalability response	Response time
	Disaster recovery time	Response time
Reliability	Service Accuracy	Failure frequency
Flexibility	Data migration	Level of data format compatibility
	Data reporting and extracting features	Level of data format compatibility
Security	Data secrecy	SLA performance
	Access control	SLA performance
Features	User-friendly navigation structure, search functionality	SLA performance
	Core features supporting process steps / activities	Correspondence between features and processes

Table 9: Quality Attributes generally classified as "dynamic."

   (a) Define attribute i's Expected performance E_i as the consumer's lower satisfaction limit;
   (b) Determine consumer's Ideal level of performance of attribute i, I_i;
   (c) Determine actual Performance of quality attribute i, P_i;

3. Determine the subjective quality measure of an attribute i as perceived by a consumer using the Modified Quality formula

$$MQ_i = -(|P_i - I_i| - |E_i - I_i|)$$

   as defined in Section 4.5.3. In order to assure consumer satisfaction of quality attribute i, influence actual performance P_i such that $MQ_i \geq 0$, which is true if P_i falls between the lower satisfaction limit (expected level) and ideal level of performance, or $E_i \leq P_i \leq I_i$.

The overall quality of a service as perceived by the consumer can be calculated by summing up the Modified Quality measures of each service attribute. If the consumer has specified certain priorities in the Service Requirements Specification phase, these can be accounted for by introducing weights w_i to the attributes' Modified Quality measures:

$$MQ = \sum(\frac{w_i}{w} MQ_i)$$

.

5.4.1 Assessment of Quality Assurance Methods

In order for quality assurance methods to be able to ensure consumer satisfaction, they need to be capable of

- keeping track of (changes in) consumer
  - expected (minimum satisfactory) and
  - ideal performance levels,

  as well as

- influencing actual performance to fall between the consumer-specified satisfaction levels

for *each* quality attribute classified as "dynamic" (as found in Table 9) in situations of changing consumer requirements of these attributes, as well as in situations of environmental fluctuations influencing these attributes.

5.5 ACCQ-methodology: Conclusion

In this chapter, the ACCQ-methodology has been defined, describing

- How to determine which aspects of cloud service quality (quality attributes) can have an influence on consumer satisfaction of cloud service quality during service consumption; and
- What capabilities are required of a service provider or assurance method to be able to assure consumer satisfaction of cloud service quality during service consumption.

This methodology can be used

- to assess an existing service provider or assurance method in its ability to assure consumer satisfaction of cloud service quality during service consumption; or

- as a framework upon which an assurance method assuring consumer satisfaction of *a specific* cloud service's quality during service consumption can be developed.

The following chapter demonstrates how the ACCQ-methodology can be used to assess an existing assurance method by applying it to the DYNAMICO assurance framework [VTM+13].

Chapter 6

Application of the ACCQ-methodology on the DYNAMICO Framework

In this chapter, the use of the ACCQ-methodology, as defined in Chapter 5, in assessing an existing assurance method is demonstrated by applying the methodology to DYNAMICO [VTM+13], an existing framework for assuring cloud service quality. In Section 6.1, the DYNAMICO framework is lined out and analyzed. In Section 6.2, the ACCQ-methodology is applied on the DYNAMICO framework in order to assess DYNAMICO's capability to assure consumer satisfaction of cloud service quality during service consumption.

6.1 DYNAMICO Analysis

Villegas et al. [VTM+13] have developed the DYNAMICO framework, "a reference model for engineering context-based self-adaptive software" [VTM+13, 266]. Its goal is to improve the engineering of self-adaptive systems from the perspective of classic control theory [VTM+13, 267]. It employs adaptation mechanisms with respect to changes in adaptation goals, as well as monitoring mechanisms with respect to changes in adaptation goals and adaptation mechanisms [VTM+13, 265], in order to "regulate the satisfaction of adaptation goals and [...] application requirements" [VTM+13, 267]. DYNAMICO employs the following strategies to improve the engineering of self-adaptive systems [VTM+13, 265]:

1. Managing *adaptation properties* and *goals* as control objectives;

2. *Separation of concerns* among feedback loops required to address control objectives over time; and

3. Managing *dynamic context* as an independent control function to preserve context-awareness in the adaptation mechanism.

Villegas et al. argue that these strategies are "crucial for governing the consistency between adaptation mechanisms and control objectives, while preserving the relevance of context monitoring of the adaptation mechanism" [VTM+13, 273]. As feedback control loops are the "cornerstone of control theory [and] provide the basis for automation in [...] computing and software engineering" [VTM+13, 270], Villegas et al. employ feedback loops on different levels of dynamics to realize the strategies mentioned previously.

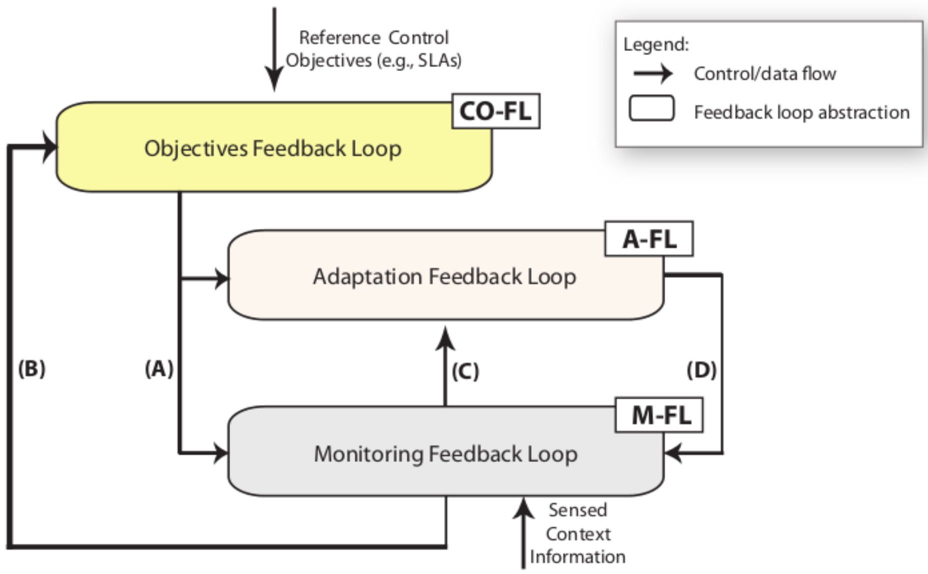

Figure 11: DYNAMICO feedback loops [VTM+13, 277].

–

6.1.1 Levels of Dynamics in Self-Adaptive Systems

Villegas et al. have identified "three levels of dynamics that must be controlled in the engineering of context-driven self-adaptive software systems" [VTM+13, 274]:

1. Management of changing control objectives by identifying changes in adaptation goals;

2. Dynamic behavior of the adaptation mechanism controlling the target system to manage changes in adaptation goals or context situations; and
3. Management of dynamic context information supporting changes in monitoring strategies at runtime.

By separating control concerns with respect to these levels of dynamics, Villegas et al. have arrived at the following three feedback loops displayed in Figure 11, respective to the above-mentioned dynamics levels (see Figure 11) [VTM^{+}13, 276-7]:

1. The *Control Objectives Feedback Loop* ("CO-FL") controls
   (a) changes in adaptation goals and
   (b) changes in monitoring requirements

   to ensure their fulfillment;
2. The *Adaptation Feedback Loop* ("A-FL") controls
   (a) the adaptive behavior of the target system and
   (b) the adaptive behavior of the adaptation mechanism

   according to control objectives while taking into account monitored context events;
3. The *Dynamic Monitoring Feedback Loop* ("M-FL") manages context information in order to preserve context relevance of the adaptation mechanism.

6.1.2 Control Objectives Feedback Loop

During service consumption, control objectives may be subject to change under the influence of for example re-negotiation of the original SLA conditions [VTM^{+}13, 279] or SLAs including dependencies on context situations [VTM^{+}13, 278]. Since control objectives or adaptation goals are subject to change by user-level negotiations at runtime, they must be addressed in a consistent and synchronized way by the adaptation mechanism and context manager [VTM^{+}13, 278]. DYNAMICO's Control Objectives Feedback Loop (CO-FL) governs such changes in control objectives. It is comprised of three components (see Figure 12):

- The *Objectives Monitor* component receives system context symptoms (both internal and external) and translates these into symptoms specific to the control objectives being governed;

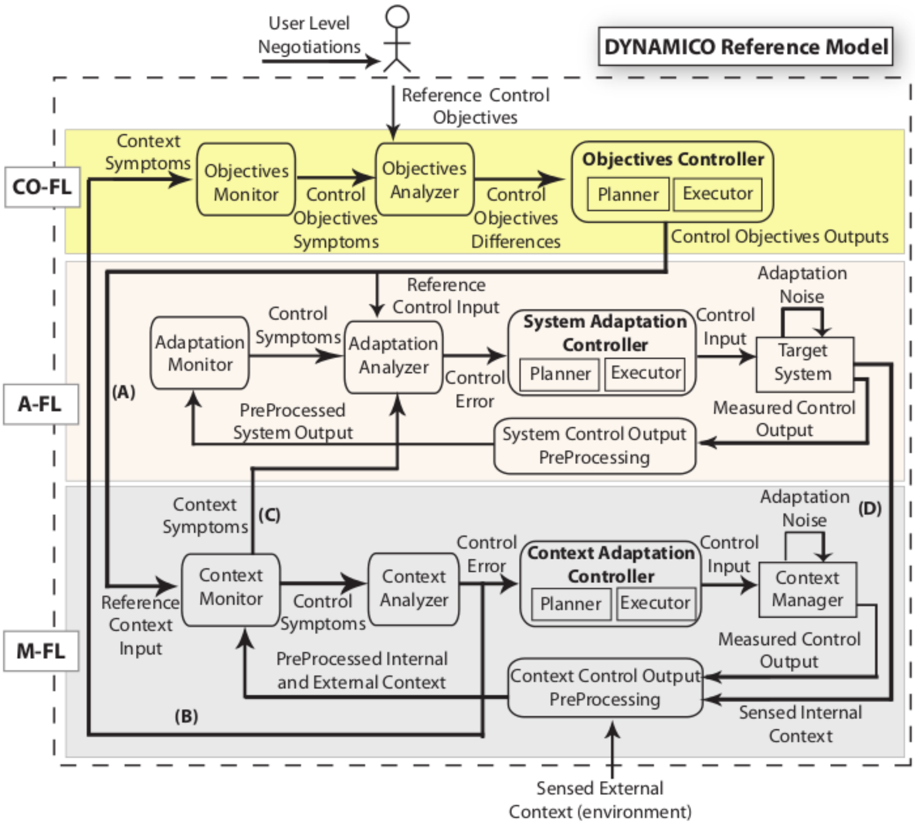

Figure 12: DYNAMICO reference model [VTM+13, 279].

- The *Objectives Analyzer* component compares the control objectives symptoms received from the Objectives Monitor to the control objectives specified by the user and analyzes differences between control objectives specified by the user and control objectives being employed by the system *at the moment*;
- The *Objectives Controller* component receives control objective differences from the Objectives Analyzer and plans and executes necessary changes in the system control objectives to overcome these differences.

Together, these three components should be capable of handling changes of control objectives resulting from user-specified control objectives changes as well as changes resulting from system context fluctuations.

6.1.3 Adaptation Feedback Loop

The Adaptation Feedback Loop (A-FL) "regulate[s] the target system's requirements satisfaction and preserve[s] the adaptation properties" [VTM+13, 280]. It measures the error in the controlled system variables respective to the reference control inputs for these variables and performs a system adaptation, if required. The A-FL is the only feedback loop in DYNAMICO that influences the target system directly. It consists of the following components (see Figure 12):

- The *System Control Output PreProcessing* component measures the target system's control output and preprocesses it for monitoring purposes;
- The *Adaptation Monitor* translates the preprocessed system output into control symptoms specific to the control objectives being governed;
- The *Adaptation Analyzer* compares the reference control input (control objectives) to system control and context symptoms and determines whether a control error is present;
- The *System Adaptation Controller* receives control errors from the Adaptation Analyzer. The Planner element of the System Adaptation Controller selects an adaptation strategy to ensure fulfillment of the control objectives being violated by the control error and orders its Executor element to perform the selected adaptation strategy upon the target system.

6.1.4 Monitoring Feedback Loop

The Monitoring Feedback Loop (M-FL) "address[es] the dynamic nature of context information [by] analyz[ing] context symptoms and facts to support the system adaptation and the management of control objectives[, while] adapt[ing] itself to support new context management requirements as the common control objectives are re-negotiated or the adaptive system evolves" [VTM+13, 281]. At its heart lies the Context Manager, which "analyze[s] context symptoms and facts to support the system adaptation and the management of control objectives [in order to] make decisions based on past, current and foreseeable future states of context[, while] adapt[ing] itself to support new context management requirements as the common control objectives are re-negotiated or the adaptive system evolves" [VTM+13, 281]. The following components enable DYNAMICO's Monitoring Feedback Loop to fulfill this function (see Figure 12):

- The *Context Control Output PreProcessing* component aggregates sensed internal context information from the target system and sensed external context information from the system's environment. It preprocesses this information using the measured control output from the context manager, which specifies the control objectives for the internal and external context;

- The *Context Monitor* compares the preprocessed internal and external context information to the reference context input containing context control objectives. The results are translated into either context symptoms (effecting the target system adaptation process) or control symptoms (indicating context manager control errors);

- The *Context Analyzer* translates control symptoms into control errors regarding either control objectives or monitoring strategies;

- The *Context Adaptation Controller* receives control errors from the Context Analyzer, based on which the controller's Planner element selects an (adapted) monitoring strategy to overcome the control error received. The controller's Executor element subsequently applies this strategy on the context manager.

6.1.5 Types of Adaptation

Villegas et al. argue that applying a separation of control concerns as applied in the DYNAMICO framework allows for three different types of adaptation; *preventive*, *corrective* and *predictive* adaptation [VTM+13, 276].

By notifying the adaptation feedback loop about context events that are likely to have an effect on the target system behavior in the future, the dynamic monitoring feedback loop is capable of *preventive* adaptation to context-related situations. Preventive adaptation handles situations that, even though they do not have an effect on the system at present, will cause an effect in the future. An example of such a situation may be the publishing of a ground-breaking scientific research report on a system in the weekend; during this weekend, access requests to this report will probably not exceed average system access requests, while it can be expected that requests will soar after the media have paid attention to its publication on Monday morning. Preventive adaptation can handle this by scaling up resources to prevent the expected rise in access requests breaking the system.

Corrective adaptation occurs when "monitoring mechanisms supporting the adaptation feedback loop detect adaptation goals being unsatisfied" [VTM+13, 277]. The most apparent example of such a situation can be found in SLA violation; if a system's accessibility, for example, is measured to lie below the threshold agreed upon in the SLA, corrective adaptation can correct this by performing a "perhaps more aggressive system reconfiguration [than the one causing adaptation goals not being met] or apply[ing] restrictive mechanisms of use to prevent the system from collapsing" [VTM+13, 277]. Corrective adaptation handles all adaptation goals defined, not just those regarding SLAs.

Predictive adaptation uses "historical information to anticipate risks of goal violation, as well as the identification of plausible symptoms that provide evidence to necessitate adaptation" [VTM+13, 278]. Such symptoms can be identified by analyzing patterns of correlated events and their effects. An event that has little effect on the system when it occurs by itself may have historically proven to have severe implications when a certain other event takes place concurrently. By analyzing and proactively reacting to such situations, predictive adaptation may prevent the system from suffering such severe implications.

6.1.6 Feedback Loop Interactions

Even though control concerns have been separated over the different feedback loops and each feedback loop achieves its control objectives independently, in order for the overall system objectives to remain fulfilled the feedback loops must cooperate by sharing information. DYNAMICO's feedback loops cooperate through interactions labeled (A), (B), (C) and (D), as shown in Figure 11 and in more detail in Figure 12. This section describes what information is shared between the independent feedback loops towards achieving their common goal.

- *Interaction (A)* provides the Monitoring Feedback loop with reference context input received from the Control Objectives Feedback Loop in the form of context manager requirements. These requirements have been identified by the CO-FL's Objectives Controller and serve as a reference for the M-FL's Context Monitor to compare the current target system's internal and external context to. Interaction (A)'s function is to "maintain [the Monitoring Feedback Loop's] relevance with respect to the actual context situation and contracted conditions [while providing the reference context input for the M-FL to] decide on context management strategies" [VTM+13, 282].
- *Interaction (B)* informs the Control Objectives Feedback Loop whenever the Monitoring Feedback Loop detects that "given the current context, the current set of control objectives should be adjusted or renegotiated dynamically" [VTM+13, 283]. The information provided to the CO-FL by Interaction (B) enables the CO-FL to make decisions about changing the system objectives to ensure the Adapta-

tion and Monitoring Feedback Loops share common control objectives [VTM+13, 283].

- *Interaction (C)* provides the Adaptation Feedback Loop's Adaptation Analyzer with relevant context symptoms necessary for the A-FL to make decisions. The M-FL identifies which context symptoms are relevant based on the current control objectives. The type of information shared through Interaction (C) varies respective to the type of adaptation employed [VTM+13, 283]. For predictive adaptation, the M-FL can inform the A-FL of the contextual events symptomatic of the necessity of future adaptation (example communications of M-FL to A-FL through Interaction (C): in cases where events X occur future adaptation will be necessary to satisfy control objective a; in cases where events Y occur no future adaptation will be necessary, unless events Z occur as well). For preventive adaptation, similar communication takes place but adaptation occurs immediately (M-FL to A-FL: in the past, events X have lead to control objective c being violated; events X are currently occurring, threatening the violation of control objective c). For corrective adaptation, context symptoms can be pushed to the A-FL by the M-FL (M-FL to A-FL: events W violate control objective e or are an indication of control objective f being violated) or pulled by the A-FL from the M-FL (A-FL to M-FL: control objective d is currently being violated; are there any context symptoms indicative of the cause of this violation?).

- *Interaction (D)* provides the Monitoring Feedback Loop with the capability to sense the target system's internal context. This enables the M-FL to "assess the system consistency after an adaptation [and] analyz[e] internal context information that characterizes the current state of system properties [in order to] provide useful information [regarding] the relationship between context symptoms, achievement of system goals and the preservation of adaptation properties" [VTM+13, 281].

6.2 DYNAMICO Assessment: Case Scenarios

In this section, the DYNAMICO quality assurance framework is assessed according to the ACCQ-methodology as defined in Chapter 5. In order to satisfy the criteria specified by this methodology, DYNAMICO must be able to handle changes in user requirements of and environment influences on the "dynamic" quality attributes found in Table 9. This implies that DYNAMICO must retain the capability to perform system adaptations such that the Modified Quality measurements (as defined in Section 5.4) of every "dynamic" quality attribute i remains equal to or greater than 0. This is the case when a quality attribute i's *actual performance* P_i lies between that attribute i's levels of *expected performance* E_i and *ideal performance* I_i, as the range between E_i and I_i can be perceived as that consumer's Zone of Tolerance for attribute i. In short, for DYNAMICO to be assessed as capable of ensuring consumer satisfaction of cloud services during their lifecycle, for each "dynamic" quality attribute i, DYNAMICO must be capable of ensuring $MQ_i \geq 0$, where $MQ_i = -(|P_i - I_i| - |E_i - I_i|)$, by ensuring $E_i \leq P_i \leq I_i$.

In the following subsections, this assessment is carried out by defining case scenarios of changing consume requirements and environment variables for *each* quality attribute classified as "dynamic", as found in Table 9, and presenting a possible reaction of DYNAMICO to each case scenario, demonstrating how DYNAMICO could assure consumer satisfaction of quality in each scenario. The case scenarios defined in the next section are designed to maximally cover their respective attribute's possible requirements variance.

6.2.1 Consumer Support

Metric: SLA performance

Changes in Consumer Requirements

Changes in support method: A consumer may wish to extend available consumer support methods to include personal support via telephone.

Adaptation:

1. The consumer changes their requirements of this attribute by specifying new adaptation objectives regarding the extended consumer support methods.
2. DYNAMICO's Monitoring Feedback Loop (M-FL) recognizes the need for new monitoring requirements regarding monitoring of the new feature, as well as new adaptation strategies incorporating this new monitoring strategy, in order to satisfy the new objectives.
   (a) If the M-FL finds these monitoring requirements and adaptation strategies to already be in place (for example because the consumer has specified them alongside the new objectives or they were already present), DYNAMICO is capable of assuring consumer satisfaction regarding this attribute with the help of the corresponding monitoring techniques and adaptation strategies (see for example case scenario "Changes in support availability" of this quality attribute).

(b) If these new monitoring requirements and adaptation strategies are not already present or have not been specified alongside the new objectives, DYNAMICO checks whether adaptation strategies exist to overcome this. This strategy may instruct DYNAMICO to provide feedback to the consumer indicating the need of specification of monitoring and adaptation capabilities respective to the new objectives, or the strategy may instruct DYNAMICO to incorporate cloud service elements providing these capabilities. By ensuring adaptation strategies and monitoring mechanisms respective to the new objectives are available, DYNAMICO can assure consumer satisfaction as specified in the new objectives with regard to this attribute. Possible prioritization of support methods may imply several adaptation strategies to be implemented pivoting on monitoring results. For example, a consumer requirement states always phone support if availability of such is less than 90%, otherwise default to email support to prevent waiting for available phone operator, with the exception of support requests labeled "urgent.")

Changes in support availability: A consumer may wish to reduce the minimum satisfactory level of consumer support availability, for example in order to reduce costs, from 90% to 40%.

Adaptation:

1. The consumer changes the requirement of this attribute by changing the respective adaptation objective to $E_{\text{supportAvailability}} = 0.4$.

2. As DYNAMICO's Monitoring Feedback Loop (M-FL) registers this objective has changed, it checks whether $P_{\text{supportAvailability}}$ falls between $I_{\text{supportAvailability}}$ and the new $E_{\text{supportAvailability}}$.

   (a) If so, the new objective is still satisfied, indicating the consumer will be satisfied with the current performance of attribute *supportAvailability.*

   (b) If the new objective is being violated, for example because $P_{\text{supportAvailability}} = .35 \leq E_{\text{supportAvailability}}$, the M-FL notifies DYNAMICO's Adaptation Feedback Loop (A-FL) of this violation.

3. If the A-FL is notified of an objective violation, it selects a suitable adaptation strategy which can influence $P_{\text{supportAvailability}}$ in such a way to fall between $E_{\text{supportAvailability}}$ and $I_{\text{supportAvailability}}$, as required for guaranteeing consumer satisfaction of this quality attribute.

(a) If no suitable adaptation strategy is present in the A-FL, the M-FL recognizes this and can reconcile this issue in a similar fashion as in Step 2b of case scenario "Changes in support method" of this attribute.

Changes in Environment Variables

New consumers: A group of new consumers (new employees) with limited to no experience using the service gains access to the service, increasing the likelihood of consumer support being relied upon.

Adaptation:

1. If an adaptation strategy in the A-FL has been implemented specifically for such cases (the possibility of such cases arising has been recognized during Design or Service Improvement phases and accounted for by implementing a relevant adaptation strategy), support can be scaled up, as dictated by the relevant adaptation strategy, at the moment the new consumers are registered with the service to prevent support availability to fall below $E_{\text{supportAvailability}}$ (preventive adaptation).
2. If such an adaptation strategy was not implemented, the system will not recognize the need for scaling up support until the $E_{\text{supportAvailability}}$ agreed upon in the SLA is being threatened with violation;
   (a) At the moment the M-FL registers $E_{\text{supportAvailability}}$ is (about to be) violated, it notifies the A-FL of the need of corrective adaptation regarding this objective / attribute.
   (b) Upon being notified by this, the A-FL will try to correct $E_{\text{supportAvailability}}$ violation by employing existing adaptation strategies, for example by scaling up of support or attempting to limit support requests.
      i. The Monitoring Feedback Loop may recognize the relationship between a large group of new consumers being registered and support requests rising shortly thereafter (after this or successive instances of similar cases) and apply predictive adaptation of support next time a similar case occurs.

6.2.2 Dynamic scalability response

Metric: Response time

Changes in Consumer Requirements

Change in resource priorities: A consumer may want to (temporarily) assign a higher priority to a certain type of resource, such as CPU, for example in order to prioritize an important service operation requiring a lot of computational power over other operations such as data read/write operations.

Adaptation:

1. The consumer informs the system of the new priority required by changing an existing adaptation objective regarding resource priorities, or by creating a new adaptation objective.
2. Based on this new or changed objective, DYNAMICO's A-FL can select an appropriate adaptation strategy to maximize the satisfaction of this objective, based on information provided by the Monitoring Feedback Loop.
3. If the M-FL recognizes a need for scaling up resources, the relevant adaptation strategy in the A-FL dictates to scale up CPU capacity first, if necessary in order to satisfy the new or changed objective.

Change in response time limits: A consumer may wish to lower the maximum allowable response time $I_{\text{scalingResponseTime}}$ or minimum tolerable response time $E_{\text{scalingResponseTime}}$.

Adaptation:

1. The consumer indicates their new requirement of this attribute by changing the relevant adaptation objective present in the CO-FL.
2. The Adaptation Feedback Loop considers whether the new objectives can be satisfied by the current adaptation strategy regarding scalability response time and selects a new strategy if necessary.
3. Based on context information from the M-FL, system adaptation according to the adaptation strategy is performed on the target system by the A-FL, if necessary in order to satisfy the new consumer requirements.

Changes in Environment Variables

Sudden unavailability of portion of resource pool: A large portion of the service's underlying infrastructure fails, resulting in unavailability of a share of resources.

Adaptation:

1. Context sensors in the Monitoring Feedback Loop register a share of resources being inaccessible, informing the Adaptation Feedback Loop of this.
2. The A-FL considers active adaptation strategies regarding this situation (such as resource priorities, service availability limits and/or backup resource pools) and performs necessary adaptations on the target system to maximize the satisfaction level of the adaptation objectives, as reflected in (weighted) aggregated Modified Quality measurements. For example, if CPU capacity satisfaction has a higher priority (larger weight) than service availability satisfaction and if satisfying $E_{\text{cpuCapacity}}$ and $E_{\text{serviceAvailability}}$ simultaneously is not immediately possible, adapt the system to satisfy $E_{\text{cpuCapacity}}$ before $E_{\text{serviceAvailability}}$).

6.2.3 Disaster recovery time
Metric: Response time

Changes in Consumer Requirements

In order for DYNAMICO to be able to assure consumer satisfaction of quality attribute Disaster recovery time, it needs to make sure proper adaptation strategies are in place *prior to* disaster occurring. This can be taken care of by adopting an adaptation objective in the spirit of "capable adaptation strategies need to be implemented regarding Disaster recovery time objectives."

Consumer changes minimum satisfactory disaster recovery time: A consumer may wish to reduce the minimum satisfactory recovery time for disaster "database corruption" to for example $E_{\text{databaseRecoveryTime}} =$ 1 day.

Adaptation:

1. The consumer changes the relevant adaptation objective $E_{\text{databaseRecoveryTime}}$ in the CO-FL to reflect their new requirements.

2. In order to comply to the adaptation objective regarding the availability of capable adaptation strategies to satisfy Disaster recovery time objectives, DYNAMICO's M-FL checks whether this objective is still satisfied in light of the new $E_{\text{databaseRecoveryTime}}$ well before $E_{\text{databaseRecoveryTime}}$ is being threatened with violation (preventive adaptation).
   (a) If existing strategies are capable of assuring satisfaction within the new satisfaction limit $E_{\text{databaseRecoveryTime}}$, all requirements for preventive adaptation in order to assure consumer satisfaction of this attribute are satisfied.
   (b) If existing strategies are insufficient regarding assuring satisfaction within the new satisfaction limit $E_{\text{databaseRecoveryTime}}$, the M-FL can adapt to this in a similar fashion as in Step 2b of case scenario "Changes in support method" of quality attribute Consumer Support.

Changes in Environment Variables

"Disaster" occurs: The system suffers a "disaster" such as database corruption.

Adaptation:

1. The M-FL recognizes the occurrence of a disaster and notifies the A-FL of this.
2. As the A-FL is being notified of disaster occurrence, it selects those adaptation strategies most applicable to the current adaptation objectives regarding Disaster recovery time in a similar fashion as in Step 2 of case scenario "Sudden unavailability of portion of resource pool" of attribute Disaster recovery time.

6.2.4 Service accuracy

Metric: Failure frequency

Changes in Consumer Requirements

Temporary increase in service accuracy required: A consumer may want to temporarily increase service accuracy, for instance during the week before an important deadline.

Adaptation:

1. The consumer indicates a requirement of increased service accuracy by for example changing the adaptation objective of maximum allowable failure frequency $I_{\text{failureFrequency}}$ from 0.05% to 0.02% and the minimum satisfactory limit of failure frequency $E_{\text{failureFrequency}}$ from 0.075% to 0.04%.

2. DYNAMICO's feedback loops are notified of the changed adaptation objectives via Interaction (A) (see Section 6.1.6):

   (a) The A-FL checks whether the system's current $P_{\text{failureFrequency}}$ lies within the range between the redefined $I_{\text{failureFrequency}}$ and $E_{\text{failureFrequency}}$, applying an appropriate adaptation strategy if required.

   (b) Additionally, the M-FL assesses whether the newly defined $I_{\text{failureFrequency}}$ and $E_{\text{failureFrequency}}$ are feasible under current context situations; if not, it may be necessary to

      i. either adopt a more aggressive adaptation policy, for example adapting the relevant context components to be more "hospitable" towards the new accuracy requirements, or

      ii. inform the CO-FL through Interaction (B) that the new accuracy requirements are overambitious and the adaptation objectives need to be redefined.

Changes in Environment Variables

Network failure: An unexpected client-side network failure has rendered the service inaccessible.

Adaptation:

1. The M-FL context monitor senses the network failure and informs the A-FL of the exact symptoms of this failure via Interaction (C).

2. The A-FL selects the most appropriate adaptation strategy to be adopted regarding the context symptoms, accuracy requirements (adaptation objectives) and for example service component prioritization.

   (a) Depending on the definition of such adaptation strategies, the A-FL might decide to run the service component with the highest priority (or specific high-priority tasks) via a backup network while attempting to replace or restore the failed infrastructure. By properly defining adaptation strategies for such events, the damage of such high-impact failures can be minimized by optimizing the outcome of the Modified Quality function

$MQ = \sum(\frac{w_i}{w} MQ_i)$ by giving priority to those attributes i whose w_i is greatest.

6.2.5 Data migration

Metric: Level of data format compatibility

Changes in Consumer Requirements

Changes in compatibility level: A consumer may want to increase the compatibility level of text document formats used within the service from 50%, meaning two separate formats are present and acceptable, to 100%, meaning all text documents present must be of the same format.

Adaptation:

1. The consumer updates the requirement by setting the relevant adaptation objective $I_{\text{formatCompatibility}} = 100\%$ (and changing the minimum tolerable limit $E_formatCompatibility$, if desired).
2. The A-FL can now analyze whether the system performs satisfactorily by using the updated Reference Control Input regarding data format compatibility provided through Interaction (A), and if necessary
   (a) apply a suitable adaptation strategy on the system. In this case this is most likely an export algorithm translating files in an undesired format to the desired format.
   (b) In case such an algorithm is not available in the system, the M-FL recognizes this through its ability to sense internal context via Interaction (D).
      i. The Reference Context Input indicates a need for translation capabilities while sensed internal context indicates such capabilities are not present. This absence of required capabilities is translated into
         - Context Symptoms provided to the A-FL through Interaction (C), enabling the A-FL to react to the current situation by for example adopting an adaptation strategy that at least forces all new text documents to be created in the desired format, thus increasing $P_{\text{formatCapbility}}$ to get closer to $I_{\text{formatCompatibility}}$; and
         - Control Symptoms informing the M-FL of the lack of capabilities to satisfy adaptation objectives, enabling the M-FL's Context Adaptation Controller to plan and execute a context adaptation strategy to overcome these

control symptoms (for example by requesting the service provider to provide additional service components providing the required capabilities, similarly to Step 2b of case scenario "Changes in support method" of attribute Consumer Support).

Changes in Environment Variables

A new data format is introduced: Due to for example a portion of the system users having "silently" migrated to a new type of office suite employing different data formats, the system is now being presented additional data formats.

Adaptation:

1. The M-FL recognizes the presence of a new data format through its context sensors and informs the rest of the system of any consequential symptoms by comparing the sensed context to relevant Reference Context Input information.

2. Depending on current adaptation objectives, appropriate actions can be performed to keep all relevant P_i between I_i and E_i;

   (a) For example, if $I_{\text{formatCompatibility}} = 50\%$ (see above in "Changes in compatibility level" under Data Migration) and only one other data format of the same type is already present, no action needs to be performed, as $P_{\text{formatCompatibility}} = 50\%$ in this case.

   (b) If $I_{\text{formatCompatibility}} = 100\%$ however, a similar adaptation may be required as in Step 2 of case scenario "Changes in compatibility level" of this attribute, or adaptation objectives need to be adapted (as will be recognized by the CO-FL through information received from the M-FL through Interaction (B)), similar to Step 2(b)ii of case scenario "Temporary increase in service accuracy required" of attribute Service accuracy.

6.2.6 Data reporting and extracting features
Metric: Level of data format compatibility

Due to the Data reporting and extracting features attribute essentially being a specific instance of Data migration (including both attributes being measurable by the same metric), the ability of satisfying this attribute can be assessed using the same case scenarios as attribute Data migration.

6.2.7 Data secrecy

Metric: SLA performance

Changes in Consumer Requirements

Changes in secrecy requirements: A consumer may want to improve data secrecy by requiring the minimum amount of time needed to crack encryption by brute force to be double of the current $I_{\text{minimumCrackingTime}}$ and/or $E_{\text{minimumCrackingTime}}$, if desired.

Adaptation:

1. The M-FL registers a change in adaptation objectives and it compares the current secrecy level P_{secrecy} to the updated objectives (in this scenario $P_{\text{secrecy}} = P_{\text{minimumEncryptionCrackingTime}}$ since no other secrecy parameters or requirements are used).
   (a) If the new objectives are not satisfied, the M-FL notifies the A-FL of this.
   (b) The A-FL then adopts an adaptation strategy capable of satisfying the new objectives, for example a strategy that doubles the complexity of encryption.
2. If the M-FL senses current adaptation strategies are incapable of satisfying the new objectives, it signals a need for adapting internal context by adding strategies capable of satisfying the objectives, for example implementing a different, stronger encryption algorithm via service composition, similar to Step 2b of case scenario "Changes in support method" of attribute Consumer support.

Changes in Environment Variables

Encryption key leaked: The M-FL senses data within the service is or can be decrypted by parties that should not be able to.

Adaptation:

1. Through information the M-FL receives from internal and external context, adaptation objectives and historical data, it deduces data secrecy has been compromised and informs the A-FL of this by providing relevant Context Symptoms through Interaction (C).
2. Based on the adaptation strategies implemented in the A-FL, the Adaptation Feedback Loop takes appropriate actions to ensure the

satisfaction of the objectives. These strategies may dictate to for example

- re-encrypt all (affected) data using a new encryption key,
- adopting entirely different encryption mechanisms

or anything in between, as indicated necessary by the relevant Context Symptoms, Reference Control Input and adaptation strategies.

6.2.8 Access control

Metric: SLA performance

Changes in Consumer Requirements

Possibilities for temporary access required: A consumer may require possibilities for temporary access to the service with certain rights (for example only access to specific, public files), for instance in order to support temporary intern employee responsibilities.

Adaptation:

1. By requiring a new type of temporary user with certain rights, the consumer introduces new adaptation objectives (for instance $I_{\text{tempUserFiletypeAccess}} = E_{\text{tempUserFiletypeAccess}} = \text{publicFilesOnly}$).
2. If not present, the M-FL has to introduce new monitoring capabilities to monitor these objectives, and the A-FL needs adaptation strategies to ensure the objectives are not violated.
3. In case this can not be realized (the system's current capabilities can not satisfy the new objectives), the M-FL can
   (a) attempt to adapt the context in such a way that those capabilities are provided (for example create a dedicated user type for this situation), or
   (b) notify the CO-FL through Interaction (B) that objectives need to be adapted or supplemented, similar to Step 2(b)ii of case scenario "Temporary increase in service accuracy required" of attribute Service accuracy.

Changes in Environment Variables

A backdoor access opportunity is being abused: The M-FL senses undesirable parties, by utilizing an unintentional "back door," have access to service components.

Adaptation:

1. The M-FL senses undesirable parties have access to service components they should not be able to access by utilizing an unintentional "back door" and notifies the A-FL of this.
2. Similarly to Step 2 of case scenario "Encryption key leaked" of this attribute (substitute "accessibility constraints" for "encryption"), the A-FL employs appropriate adaptation strategies to maximize satisfaction of relevant adaptation objectives.

6.2.9 User-friendly navigation structure, search functionality
Metric: SLA performance

Changes in Consumer Requirements

Consumer redefines minimum satisfaction limit of user- friendliness of navigation structure: The consumer may wish to redefine minimum satisfactory user-friendliness of navigation structure to "user- configurable" navigation structure.

Adaptation:

1. The consumer changes their requirements of this attribute to $E_{\text{navigationStructure}} = \textit{userConfigurable}$.
2. The M-FL registers the changed $E_{\text{navigationStructure}}$ and checks whether the current level of $P_{\text{navigationStructure}}$ falls between the new $E_{\text{navigationStructure}}$ and existing $I_{\text{navigationStructure}}$ levels.
   (a) If so, the new adaptation objective is being satisfied, allowing for the assumption that the consumer is satisfied with the current quality of this attribute.
   (b) If $P_{\text{navigationStructure}} < E_{\text{navigationStructure}}$ or $P_{\text{navigationStructure}} > I_{\text{navigationStructure}}$, the M-FL notifies the A-FL of this.
3. Upon notification of objective violation, the A-FL adopts a suitable adaptation strategy capable of influencing $P_{\text{navigationStructure}}$ in such a way that $E_{\text{navigationStructure}} \leq P_{\text{navigationStructure}} \leq I_{\text{navigationStructure}}$.
   (a) If no adaptation strategy capable of satisfying the new objective is available, this can be overcome in a similar fashion as in Step

2b of case scenario "Changes in support method" of attribute Consumer support.

Changes in Environment Variables

A service feature previously represented in search results has been removed from the service: A service component previously offered as a result in searches has been removed from the service, potentially resulting in consumers not being presented with the search results they expected.

Adaptation:

1. The consumer indicates the presented search results did not provide the expected result.
2. The M-FL registers a violation of $E_{\text{searchFunctionality}}$ and notifies the A-FL of this.
3. Upon notification of objective violation, the A-FL attempts to correct this violation by checking if suitable adaptation strategies are available which are capable of satisfying the violated objective $E_{\text{searchFunctionality}}$.
   (a) Since the system has no knowledge of the feature the consumer expected to be presented with in the search results (due to that feature not existing), it is impossible for the A-FL to deduce exactly what the consumer requirement of $E_{\text{searchFunctionality}}$ was. As overcoming this would involve adaptation strategies attempting to distinguish *unknown consumer requirements*, and such strategies can not be expected to be automatically incorporated into the service as they involve complicated heuristics-like algorithms, DYNAMICO can not be assumed to be able to assure consumer satisfaction of quality attributes involving unknown consumer requirements.

6.2.10 Core features supporting process steps / activities

Metric: Correspondence between features and processes

Changes in Consumer Requirements

Increase of required correspondence between features and processes: The consumer may want to increase the correspondence between features and processes from $E_{\text{featureProcessesCorrespondence}} = global$, meaning integral features' function results must correspond to the respective business processes' results, to $E_{\text{featureProcessesCorrespondence}} = literal$, meaning each step of a service feature must correspond exactly to its respective business process step.

Adaptation:

1. The M-FL registers the changed adaptation objective and employs the respective monitoring mechanism, which compares the business process model to the service feature model.
   (a) If the monitoring mechanism finds that $P_{\text{featureProcessesCorrespondence}} \geq E_{\text{featureProcessesCorrespondence}}$, the new objectives are being satisfied, allowing for the assumption that the consumer is satisfied with the quality of this attribute.
   (b) If $P_{\text{featureProcessesCorrespondence}} < E_{\text{featureProcessesCorrespondence}}$, the M-FL notifies the A-FL of this objective violation.
2. The A-FL adopts a suitable adaptation strategy capable of influencing $P_{\text{featureProcessesCorrespondence}}$ in such a way for it to fall between $E_{\text{featureProcessesCorrespondence}}$ and $I_{\text{featureProcessesCorrespondence}}$ by changing service feature steps to correspond to business process steps.
   (a) If no suitable adaptation strategies are available, the A-FL can correct this in a similar fashion as in Step 2b of case scenario "Changes in support method" of attribute Consumer support.

Changes in Environment Variables

The business process has been changed: The way the business' processes are structured changes dramatically, for example its structure is changed from hierarchical to teamwork-based.

Adaptation:

1. The M-FL registers a change in the service environment and concludes that the business model present in the system does not accurately reflect the actual business' processes any more, and notifies the A-FL of this.
2. The A-FL then attempts to adapt the business model in such a way that it represents the actual business processes by employing existing adaptation strategies regarding adapting the business process model.
   (a) Since the business was previously always structured hierarchically, existing adaptation strategies are only capable of performing adaptations on a hierarchical business model, and have no

knowledge of (the existence of) teamwork-based processes. Since such a structure might involve a new business structure, it can not be assumed that such adaptation strategies exist, and they can therefor not be automatically incorporated into the service. For this reason, DYNAMICO can not be assumed to be capable of assuring consumer satisfaction of quality attribute Core features supporting business process / steps and activities without some form of heuristic algorithm or human intervention in the form of model language re-specification to include teamwork-based structures, in this case.

6.3 DYNAMICO Assessment: Conclusion

As demonstrated in this section, DYNAMICO is capable of assuring consumer satisfaction of “dynamic” quality attributes Consumer support, Dynamic scalability response, Disaster recovery time, Service accuracy, Data migration, Data reporting and extracting features, Data secrecy and Access control under situations of changing consumer requirements of these attributes as well as environmental fluctuations influencing these attributes. However, for attributes User-friendly navigation structure, search functionality and Core features supporting process steps / activities, DYNAMICO was not able to assure consumer satisfaction in every case scenario. This inability is caused by the system being required to uncover unknown consumer requirements or unknown environment states, requiring elements of human interaction and / or heuristic algorithms in order to be able to assure satisfaction of these attributes. Such elements cannot be assumed to be readily available for incorporation in such cloud service systems, leading to the conclusion that further research is necessary on how to incorporate such elements, if possible, into assurance methods for consumer satisfaction of cloud service quality.

Chapter 7

Related Work

This chapter provides an overview of research complimentary to the research performed in this thesis.

7.1 Related Work: Cloud Computing

Khalid [Kha10] has addressed issues in deploying Cloud Services in small businesses. Hosono et al. [HKH+09] have developed a framework for co-creating business values between providers and consumers of IT services. Wang and Xu [WX09] have incorporated the service value concept into the cloud service lifetime cycle. Jegadeesan and Balasubramaniam [JB09] have developed a method to support variability of enterprise cloud services.

Schroeter et al. [SMM+12] propose a dynamic cloud service configuration management method. Bai et al. [BLC+11] have surveyed representative approaches and typical tools for integrative cloud testing. Wang et al. [WZWQ10] have developed a trust-based service quality evaluation model. Chakraborty and Roy [CR12] have developed a framework for estimating cloud service trustworthiness based on SLAs. Bochicchio et al. [BL11] have developed a model for cloud service contract management. Ma et al. [MPT05] have performed an exploratory study into factors of cloud service quality.

Verheecke et al. [VCJ03] have developed the Web Service Management Layer for dynamic selection and integration of services into an application, client-side management of the service and support for rules that govern the selection, integration and composition of services.

7.1.1 SLA Compliance

Andrzejak et al. [AKY10] have developed a decision model for cloud computing under SLA constraints, optimizing costs, performance and reliability. Chazalet [Cha10a][Cha10b] has developed an architectural approach to service level checking. Parkin and Morgan [PM12] have done research in the field of reusable SLA monitoring capabilities. Sahai et al. [SMS+02] have proposed an automated SLA monitoring engine for cross-enterprise situations. Bose et al. [BPR+11] have examined SLA management from a service provider's perspective. Buco et al. [BCL+04] have developed a business objectives-based SLA management system. Comuzzi et al. [CKSY09] have examined the relationship between establishing and monitoring SLAs. Haq et al. [HBS10] have developed a systematic approach to managing SLAs from the resource, infrastructure and business perspectives. Lee et al. [LLK09] have defined QoS metrics for Software-as-a-Service evaluation. Theilmann et al. [THK+10] have presented a reference architecture for a multilevel SLA management framework. Xiong and Perros [XP09] have examined the relationship between maximal number of customers, minimal service resources and the highest level of services from a provider's perspective. Yfoulis and Gounaris [YG09] have approached honoring SLAs from a control perspective.

7.2 Related Work: Consumer Satisfaction

Sureshchadar et al. [SRA02], Cronin et al. [CJBH00], Ha [Ha06] and Udo et al. [UBK10] have examined the relationship between service quality and consumer satisfaction. Boulding et al. [BKSZ93] have done research on the relationship between consumer expectations and behavior. Goodwin and Ross [GR92] have performed research on consumer responses to service failure.

Chapter 8

Summary and Conclusion

In the relatively new Cloud Computing paradigm, required IT resources are essentially rented on demand and accessed remotely, in theory allowing for consumers to pay for, use and adapt the cloud service during its runtime as desired. While methods exist that attempt to deal with dynamic service configuration and assurance of its quality, most of these are mainly designed with provider profit as its main goal, as becomes apparent from the emphasis on Service Level Agreement violation prevention in order to minimize SLA violation fees for the provider. An example of such a provider-oriented quality assurance method is Qu4DS, described in Section 3.3.

In this thesis, the ACCQ-methodology (Assuring Consumer satisfaction of Cloud service Quality) has been developed, which prescribes what is required for a service provider or assurance method to assure consumer satisfaction of cloud service quality during service consumption. This methodology has been developed by first examining and defining Cloud Computing, Cloud Service quality and Consumer satisfaction theories within the relevant domain. Subsequently, attributes were extracted that are relevant towards consumers' satisfaction with cloud service quality in situations where a service's internal or external context fluctuates during consumption. The capabilities required of service providers or assurance methods to assure satisfaction of these attributes.

In order to demonstrate its use, the ACCQ-methodology was applied on the DYNAMICO assurance framework [VTM+13]. In this application, DYNAMICO was analyzed and assessed on its ability to satisfy each of the extracted quality criteria regarding consumer satisfaction in situations of fluctuating internal and external service environments.

8.1 Contribution

The service-like nature of Cloud Computing allows for consumers to dynamically adapt the system to their current wishes during cloud service consumption. Service Level Agreements are negotiated between service providers and consumers to document the specifications of the service to be delivered and consumed. It is the service provider's responsibility to ensure the service operates within the limits specified in the SLA on penalty of violation fees. Current methods dealing with dynamic service configuration and quality assurance have been developed towards the goal of minimizing violation fees for the service provider by attempting to ensure service quality is maintained within the bounds specified by the SLA.

The question raised by such methods of quality assurance is whether consumer satisfaction can be guaranteed by merely assuring technical quality aspect in order to prevent SLA violation, or whether consumer satisfaction of cloud service quality is affected by more aspects than SLA violation (prevention) alone? Consequently, what capabilities are required from a cloud service quality assurance method in order for it to be able to ensure *consumer satisfaction of* a cloud service's quality, as opposed to ensuring quality as specified in an SLA? Are existing assurance methods capable of satisfaction assurance from the consumer's perspective?

8.2 Research Method

In order to answer these questions, both cloud computing and consumer satisfaction theories have been examined in order to extract aspects relating to consumer perception of service quality. By applying consumer satisfaction theory on cloud service quality, a set of cloud service quality attributes were identified that can influence consumer satisfaction of service quality. The ACCQ-methodology developed in this thesis prescribes how to distinguish the cloud service quality attributes that can influence consumer satisfaction of cloud service quality *during service consumption* as well as what is required of a service provider or assurance method to ensure consumer satisfaction during cloud service consumption. This methodology can be used either to assess existing assurance methods in their capability to ensure consumer satisfaction of cloud service quality in general or for a specific cloud service, or it can be used as a reference model to support developing an assurance method. In order to demonstrate the use of the ACCQ-methodology, it has been applied on DYNAMICO [VTM+13], a framework for service quality assurance.

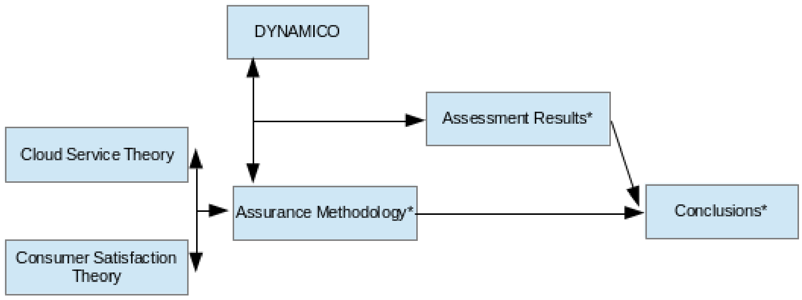

Figure 13: Research Method (* represent original contributions).

8.3 Cloud Computing

Cloud Computing technologies can be provided on an Infrastructure-, Platform-, and/or Software-as-a-Service basis, each layer offering virtualized services where the consumer has virtualized and abstracted access to the necessary resources through standard networking protocols. The U.S. National Institute of Standards and Technology has identified 5 essential cloud computing characteristics, namely on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. The Service Provider and the Consumer negotiate on the composition and acceptance criteria of the service to be delivered, laid down in a Service Level Agreement (SLA). During service consumption, a service is delivered to the consumer on demand, quality is being monitored, maintenance is performed and monetary compensation is provided. Continuous measurement and improvement of service levels, technology employed and process efficiency and effectiveness should be performed throughout a service's lifecycle in order to meet the consumer's specifications and needs. Dimensions of Cloud Computing influencing the quality of a service include Rapport (quality of support), Responsiveness, Reliability, Flexibility, Security and Features. Each dimension includes a subset of indicators or attributes of service quality. Additionally, attributes such as provider reputation, required skills, migration process complexity, pricing tariff, costs compared to internal solution and consumer support can have an influence on the consumer's perception of service quality.

8.4 Consumer Satisfaction

A consumer's response to their judgment of the level of fulfillment provided by a service determines that consumer's level of satisfaction of that service. Fulfillment involves an outcome (performance) and a referent to compare this outcome to (expectations). Expectations can be classified by level of desire, ranging from intolerable ("My expectation of intolerable performance of feature i is $intolerablePerformance_i = 40\%$") to ideal ("My expectation of ideal performance of feature i is $idealPerformance_i = 99\%$ or $I_i = 99\%$") expectations. The zone between the minimum tolerable and ideal levels is known as the Zone of Tolerance; consumers' expected outcome E_i of the performance P_i of i usually falls within this zone. The comparison between consumer expectations and service performance results in an expectation-performance discrepancy. The difference between expectation and performance is known as disconfirmation: positive disconfirmation occurs when performance is higher than expected, negative disconfirmation when performance is lower than expected. Confirmation of expectations occurs when performance is equal to the consumer's expectations. However, positive disconfirmation does not necessarily lead to consumer satisfaction, as a consumer may have a very negative expectation of performance (for example due to low provider reputation); a slightly better performance than expected results in positive disconfirmation but may not be good enough to reach that consumer's minimum satisfaction limit. Assuming a consumer would prefer an ideal product, the overall quality of a service can be measured by collectively comparing the actual performance of all relevant features to the ideal levels of these features.

8.5 ACCQ-methodology

8.5.1 ACCQ: Classification of Attributes

The quality attributes identified in Chapter 3 have been categorized based on the consideration of whether consumer requirements of these attributes are generally deliberate design choices made prior to service consumption or whether requirements can be expected to change during the service consumption phase. For those attributes classified as deliberate design choices, changing their requirements during service runtime would alter the nature of the service in such a manner that it can be regarded as an inherently different service. These attributes will be classified as "design choice" attributes in respect to their influence on consumer satisfaction. For example, by changing the minimum satisfactory requirement for attribute Configuration features from "minimal" (only basic configuration features) to "high," the service changes from relatively uniform across all users to a highly diverse service. By contrast, changes in requirements of attributes that can be expected to change during the consumption phase result in the service behaving differently, but its inherent nature staying intact. These attributes will be classified as "dynamic" attributes in respect of their influence on consumer satisfaction. For example, changes in the requirements for attribute Service Accuracy will result in the service providing similar functionality, but under different failure frequency constraints: the minimum satisfactory value "high" of quality attribute "Service accuracy" results in a highly accurate service, while specifying the same attribute's minimum satisfactory value as "low" results in a less accurate service, but the function and nature of the service remain the same. Similar changes in requirements during service consumption can therefor reasonably be expected, while such changes in requirements of "design choice" attributes cannot reasonably be expected during service consumption as such changes would require redesigning the service. The result of this classification of quality attributes in the general case can be found in Chapter 5, specifically in Table 9. It is important to note that, when applying this methodology to a specific cloud service, the applicable quality attributes must be correctly classified in this manner, as attributes may be classified differently between services. For example, if a consumer requires a service to support changes in levels of configuration during service consumption, attribute Configuration features will be a "dynamic" attribute and needs to be classified as such, even though *in the general case* it is classified as a "design choice" attribute.

8.5.2 ACCQ: Assuring Satisfaction

As demonstrated in Chapter 4, consumer satisfaction of service quality encompasses actual performance of a service and/or attributes thereof, as well as consumers' levels of expected and ideal performance. As a result, methods attempting to assure consumer satisfaction of a cloud service's quality must be able to keep track of specifications of and fluctuations in all three of these performance aspects for each quality attribute classified as "dynamic". By defining a consumer's expected performance level to reflect that specific consumer's minimum satisfactory performance level, it is assumed that satisfaction can be assured by keeping actual performance above or equal to a consumer's expected performance level and below or equal to their ideal performance level.

Therefor, the capability of a method to assure consumer satisfaction of cloud service quality can be assessed by

- examining its capability to keep track of (changes in) consumer
  - expected (minimum satisfactory) and
  - ideal performance levels

  of each quality attribute affecting consumer satisfaction during service consumption, as well as

- the ability to influence actual performance of each of these attributes to fall between the consumer-specified satisfaction levels.

8.6 Application of ACCQ-Methodology

In order to demonstrate one of the uses of the consumer satisfaction assurance methodology developed in this thesis, the ACCQ-methodology was applied to assess DYNAMICO [VTM+13], a framework for cloud service quality assurance, in its ability to assure consumer satisfaction of cloud service quality during service consumption.

8.6.1 DYNAMICO

DYNAMICO is a reference model for engineering context-based self-adaptive software, employing

- adaptation mechanisms with respect to changes in adaptation goals;
- mechanisms monitoring changes in adaptation mechanisms; and
- mechanisms regulating the satisfaction of adaptation goals.

This is realized by separating the concerns of adaptation goals, adaptation mechanisms and monitoring mechanisms over three independent feedback loops. Interactions between the feedback loops provide each with relevant information. The separation of concerns in this manner allows for three types of adaptation: in situations where monitoring mechanisms detect adaptation objectives being unsatisfied, corrective adaptation mechanisms are activated to ensure objectives can be met. By identifying context situations that will have an effect on the target system in the future, the system is capable of supporting preventive adaptation. By utilizing historical information to anticipate risks of goal violation, predictive adaptation can be applied [VTM+13].

8.6.2 DYNAMICO Assessment: Case Scenarios

In accordance with the ACCQ-methodology, DYNAMICO was assessed in

- its ability to track (changes in) consumer requirements of minimum satisfactory and ideal levels of performance of quality attributes classified as "dynamic" in the general case, as well as

- its ability to adapt actual performance of quality attributes classified as "dynamic" under circumstances of
  - changes in consumer requirements of minimum satisfactory and ideal levels of attributes; as well as
  - fluctuations in the service's external environment.

This assessment was performed based on case scenarios of changing consumer requirements and environment variables for each quality attribute.

Example scenarios for fluctuations in each quality attribute affecting consumer satisfaction and DYNAMICO's possible reactions to these scenarios are worked out in Section 6.2. Based on these scenarios, it can be concluded that DYNAMICO is capable of ensuring consumer satisfaction of quality attributes Consumer support, Dynamic scalability response, Disaster recovery time, Service accuracy, Data migration, Data reporting and extracting features, Data secrecy and Access control. However, for the quality attributes User-friendly navigation structure / search functionality and Core features supporting process steps / activities, scenarios were encountered in which DYNAMICO was unable to influence consumer satisfaction of these attributes.

8.7 Conclusion

Consumer satisfaction was found to entail more than SLA performance, as described in Chapter 5. After a service has been designed and transgressed into the consumption phase, certain quality attributes, as found in Table 5, have an influence on consumer satisfaction of service quality, in general. For each implementation of an actual service, however, this list might not be complete or accurate due to the nature of that service. For example, if a service does not include consumer support, performance of the quality attribute "Consumer support" has no influence on the consumer's perception of satisfaction. Likewise, if a service is required by the consumer to have dynamic configuration options (the consumer would like to reserve the possibility to change the service from non-configurable to configurable in certain aspects *during service consumption*) and this possibility has been agreed upon with the service provider in the negotiation phase, the attribute "Configuration options" *does* have an influence on consumer satisfaction, even though it is not included in Table 5. However, in such cases this will have been explicitly specified in an SLA, while proper adaptation goals and mechanisms as well as minimum satisfactory and ideal performance limits should have been defined to accommodate this service feature, which in turn would allow assurance methods such as DYNAMICO to ensure consumer satisfaction regarding such attributes nonetheless.

Vital to satisfaction assurance methods is knowledge of potential values for adaptation goals, ideal and minimum satisfactory performance limits, environmental influences on these values, and possible techniques (adaptation mechanisms) to ensure the actual performance of attributes influencing consumer satisfaction of a specific service's quality remain within specified satisfaction limits.

A system having such knowledge becomes problematic in situations where unforeseeable fluctuations in consumer requirements or environmental variables take place. In situations where unforeseeable fluctuations in consumer requirements or environmental variables take place, it becomes problematic for a system to have access to this "vital" knowledge. Such unpredictable fluctuations might arise in cases of for example business process innovation, where human intervention is necessary to inform the system of the new process, as it is impossible for the system to have had knowledge of the new process and adapt to it accordingly, since the system's designers did not have knowledge of such a process even being possible before it was invented. Another example of unforeseeable fluctuations is consumer requirements unexpectedly changing to something previously unfathomable; it is impossible for a system to ensure satisfaction of something it does not realize it should ensure satisfaction *of*. An example of this is ensuring satisfaction with search functionalities: a system can measure how often it presents the option the consumer was looking for as the first option, it can measure how often it is unable to present the option the consumer was looking for at all, but it can not with certainty deduce what the option was that the consumer was looking for. If it could have, it would have presented that option in the first place. Such situations require at least a degree of (corrective) human interaction with the system or might be solved using artificial intelligence (heuristic) algorithms.

8.8 Future Work

In order to be able to assure consumer satisfaction of any quality attribute of cloud services, future research is needed concerning determining unknown consumer requirements, for instance by incorporating heuristic algorithms into a cloud service.

The ACCQ-methodology as developed in this thesis may be extended and operationalized for the use of rating or comparing cloud service quality satisfaction assurance methods.

Bibliography

[AKY10] Artur Andrzejak, Derrick Kondo, and Sangho Yi. Decision model for cloud computing under SLA constraints. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2010 IEEE International Symposium on*, pages 257–266. IEEE, 2010.

[Arr10] Valeri Arraj. ITIL: The Basics. White paper, UK Office of Government Commerce, May 2010.

[BCL+04] Melissa J. Buco, Rong N. Chang, Laura Z. Luan, Christopher Ward, Joel L. Wolf, and Philip S. Yu. Utility computing SLA management based upon business objectives. *IBM Systems Journal*, 43(1):159–178, 2004.

[BKH11] Alexander Benlian, Marios Koufaris, and Thomas Hess. Service Quality in Software-as-a-Service: Developing the SaaS-Qual Measure and Examining Its Role in Usage Continuance. *Journal of Management Information Systems*, 28(3):85 – 126, 2011.

[BKSZ93] William Boulding, Ajay Kalra, Richard Staelin, and Valarie A. Zeithaml. A dynamic process model of service quality: from expectations to behavioral intentions. *Journal of Marketing Research*, 30(1):7–27, 1993.

[BL11] Mario A. Bochicchio and Antonella Longo. Modelling contract management for cloud services. In *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, pages 332–339. IEEE, 2011.

[BLC+11] Xiaoying Bai, Muyang Li, Bin Chen, Wei-Tek Tsai, and Jerry Gao. Cloud testing tools. In *Service Oriented System Engineering (SOSE), 2011 IEEE 6th International Symposium on*, pages 1–12. IEEE, 2011.

[BPR+11] Sumit Bose, Anjaneyulu Pasala, Dheepak Ramanujam, Sridhar Murthy, and Ganesan Malaiyandisamy. SLA management in cloud computing: A service provider's perspective. *Cloud Computing: Principles and paradigms, Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Eds. New Jersey, USA: John Wiley & Sons*, pages 413–436, 2011.

[Cha10a] Antonin Chazalet. Service level agreements compliance checking in the cloud computing: architectural pattern, prototype, and validation. In *Software Engineering Advances (ICSEA), 2010 Fifth International Conference on*, pages 184–189. IEEE, 2010.

[Cha10b] Antonin Chazalet. Service level checking in the cloud computing context. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 297–304. IEEE, 2010.

[CHR+07] Alison Cartlidge, Ashley Hanna, Colin Rudd, Ivor Macfarlane, John Windebank, and Stuart Rance. An Introductory Overview of ITIL V3. High-level overview, UK Office of Government Commerce, 2007.

[CJBH00] J. Joseph Cronin Jr., Michael .K Brady, and G. Tomas M. Hult. Assessing the effects of quality, value, and customer satisfaction on consumer behavioral intentions in service environments. *Journal of retailing*, 76(2):193–218, 2000.

[CKSY09] Marco Comuzzi, Constantinos Kotsokalis, George Spanoudakis, and Ramin Yahyapour. Establishing and monitoring SLAs in complex service based systems. In *Web Services, 2009. ICWS 2009. IEEE International Conference on*, pages 783–790. IEEE, 2009.

[Com08] Computer Aid, Inc. *Service Management For Application Support*, volume 1. 2008.

[CR12] Sudip Chakraborty and Krishnendu Roy. An SLA-based framework for estimating trustworthiness of a cloud. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on*, pages 937–942. IEEE, 2012.

[FGP+10] Stefano Ferretti, Vittorio Ghini, Fabio Panzieri, Michele Pellegrini, and Elisa Turrini. QoS–aware clouds. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 321–328. IEEE, 2010.

[FPP11] André Lage Freitas, Nikos Parlavantzas, and J.-L. Pazat. Cost reduction through SLA-driven self-management. In *Web Services (ECOWS), 2011 Ninth IEEE European Conference on*, pages 117–124. IEEE, 2011.

[FPP12] André Lage Freitas, Nikos Parlavantzas, and J.-L. Pazat. An integrated approach for specifying and enforcing SLAs for cloud services. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 376–383. IEEE, 2012.

[Fre12] André Lage Freitas. *Autonomous Service Execution Driven by Service-Level Agreements.* Doctoral thesis, 2012.

[GR92] Cathy Goodwin and Ivan Ross. Consumer responses to service failures: influence of procedural and interactional fairness perceptions. *Journal of Business Research*, 25(2):149–163, 1992.

[Ha06] Hong-Youl Ha. An integrative model of consumer satisfaction in the context of e-services. *International Journal of Consumer Studies*, 30(2):137–149, 2006.

[HBS10] Irfan Ul Haq, Ivona Brandic, and Erich Schikuta. SLA validation in layered cloud infrastructures. In *Economics of Grids, Clouds, Systems, and Services*, pages 153–164. Springer, 2010.

[HKH+09] Shigeru Hosono, Ayako Kuno, Masahiro Hasegawa, Tatsunori Hara, Yoshiki Shimomura, and Tamio Arai. A framework of co-creating business values for IT services. In *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*, pages 167–174. IEEE, 2009.

[HZ13] Hao Hu and Jianlin Zhang. The Evaluation System for Cloud Service Quality Based on SERVQUAL. In *Proceedings of the 2012 International Conference on Information Technology and Software Engineering*, volume 210 of *Lecture Notes in Electrical Engineering*, pages 577 – 584. Springer Berlin Heidelberg, 2013.

[JASZ11] Si-Yuan Jing, Shahzad Ali, Kun She, and Yu Zhong. State-of-the-art research study for green cloud computing. *The Journal of Supercomputing*, pages 1 – 24, 2011.

[JB09] Harshavardhan Jegadeesan and Sundar Balasubramaniam. A method to support variability of enterprise services on the cloud. In *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*, pages 117–124. IEEE, 2009.

[JFY09] Karuna P. Joshi, Tim Finin, and Yelena Yesha. Integrated lifecycle of IT services in a cloud environment. In *Proceedings of the Third International Conference on the Virtual Computing Initiative*, 2009.

[Kha10] Ammar Khalid. Cloud computing: Applying issues in small business. In *Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on*, pages 278–281. IEEE, 2010.

[KMA10] Philip Koehler, Dan MA, and Arun Anandasivam. Cloud services from a consumer perspective. Americas Conference on Information Systems (AMCIS), 2010.

[LKN+09] Alexander Lenk, Markus Klems, Jens Nimis, Stefan Tai, and Thomas Sandholm. What's inside the Cloud? An architectural map of the Cloud landscape. In *Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing*, pages 23 –31. IEEE Computer Society, 2009.

[LLK09] Jae Yoo Lee, Jung Woo Lee, and Soo Dong Kim. A quality model for evaluating software-as-a-service in cloud computing. In *Software Engineering Research, Management and Applications, 2009. SERA'09. 7th ACIS International Conference on*, pages 261–266. IEEE, 2009.

[MBS12] Michael Maurer, Ivona Brandic, and Rizos Sakellariou. Self-adaptive and resource-efficient SLA enactment for cloud computing infrastructures. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 368–375. IEEE, 2012.

[Men02] D.A. Menascé. QoS issues in Web services. *Internet Computing*, 6(6):71 – 75, December 2002.

[MG12] Peter Mell and Timothy Grance. The NIST Definition of Cloud Computing. Technical Report Special Publication 145-800, National Institute of Standards and Technology, U.S. Department of Commerce, September 2012.

[Mil77] John A. Miller. *Studying Satisfaction, Modifying Models, Eliciting Expectations, Posing Problems, and Making Meaningful Measurements*, pages 72 – 91. Marketing Science Institute, Cambridge, MA, USA, 1977.

[MPT05] Qingxiong Ma, J. Michael Pearson, and Suresh Tadisina. An exploratory study into factors of service quality for application service providers. *Information & Management*, 42(8):1067–1080, 2005.

[Oli10] R.L. Oliver. *Satisfaction: A Behavioral Perspective on the Consumer*. Sharpe, M.E., 2010.

[OW87] R.L. Oliver and R.S. Winer. A framework for the formation and structure of consumer expectations: Review and propositions. *Journal of Economic Psychology*, 8(4):469–499, December 1987.

[PM12] Simon E. Parkin and Graham Morgan. Toward reusable SLA monitoring capabilities. *Software: Practice and Experience*, 42(3):261–280, 2012.

[PRS09] Pankesh Patel, Ajith Ranabahu, and Amit Sheth. Service level agreement in cloud computing. In *Cloud Workshops at OOPSLA*, 2009.

[PZB85] A. Parasuraman, Valarie A. Zeithaml, and Leonard L. Berry. A conceptual model of service quality and its implications for future research. *The Journal of Marketing*, pages 41–50, 1985.

[PZB88] A. Parasuraman, Valarie A. Zeithaml, and Leonard L. Berry. SERVQUAL: A multiple-item scale for measuring consumer perceptions of service quality. *Journal of Retailing*, 64(1):12 – 40, 1988.

[PZM05] A. Parasuraman, Valarie A. Zeithaml, and Arvind Malhotra. ES-QUAL a multiple-item scale for assessing electronic service quality. *Journal of Service Research*, 7(3):213–233, 2005.

[SL12] Sherif Sakr and Anna Liu. SLA-based and consumer-centric dynamic provisioning for cloud databases. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 360–367. IEEE, 2012.

[SMM+12] Julia Schroeter, Peter Mucha, Marcel Muth, Kay Jugel, and Malte Lochau. Dynamic configuration management of cloud-based applications. In *Proceedings of the 16th International Software Product Line Conference-Volume 2*, pages 171–178. ACM, 2012.

[SMS+02] Akhil Sahai, Vijay Machiraju, Mehmet Sayal, Aad van Moorsel, and Fabio Casati. Automated SLA monitoring for web services. In *Management Technologies for E-Commerce and E-Business Applications*, pages 28–41. Springer, 2002.

[Sos11] Barry Sosinsky. *Cloud Computing Bible.* Wiley Publishing, 2011.

[SRA02] G.S. Sureshchandar, Chandrasekharan Rajendran, and R.N. Anantharaman. The relationship between service quality and customer satisfaction – a factor specific approach. *Journal of services marketing*, 16(4):363–379, 2002.

[Tea93] R. Kenneth Teas. Expectations, performance evaluation, and consumers' perceptions of quality. *Journal of Marketing*, 57(4):18 – 34, October 1993.

[THK+10] Wolfgang Theilmann, Jens Happe, Costas Kotsokalis, Andy Edmonds, Keven Kearney, and J. Lambea. A reference architecture for multi-level SLA management. *Journal of Internet Engineering*, 4(1), 2010.

[UBK10] Godwin J. Udo, Kallol K. Bagchi, and Peeter J. Kirs. An assessment of customers'e-service quality perception, satisfaction and intention. *International Journal of Information Management*, 30(6):481–492, 2010.

[VCJ03] Bart Verheecke, María Agustina Cibrán, and Viviane Jonckers. AOP for dynamic configuration and management of web services. In *Web Services-ICWS-Europe 2003*, pages 137–151. Springer, 2003.

[VTM+13] Norha M. Villegas, Gabriel Tamura, Hausi A. Müller, Laurence Duchien, and Rubby Casallas. DYNAMICO: A reference model for governing control objectives and context relevance in self-adaptive software systems. In *Software Engineering for Self-Adaptive Systems II*, pages 265–293. Springer, 2013.

[WX09] Zhongjie Wang and Xiaofei Xu. SVLC: service value life cycle model. In *Cloud Computing, 2009. CLOUD'09. IEEE International Conference on*, pages 159–166. IEEE, 2009.

[WZWQ10] Shou-Xin Wang, Li Zhang, Shuai Wang, and Xiang Qiu. A cloud-based trust model for evaluating quality of web services. *Journal of Computer Science and Technology*, 25(6):1130–1142, 2010.

[XP09] Kaiqi Xiong and Harry Perros. Service performance and analysis in cloud computing. In *Services-I, 2009 World Conference on*, pages 693–700. IEEE, 2009.

[YG09] Christos A. Yfoulis and Anastasios Gounaris. Honoring SLAs on cloud computing services: a control perspective. In *Proceedings of EUCA/IEEE European Control Conference*, 2009.

[ZBP91] Valarie A. Zeithaml, Leonard L. Berry, and A Parasuraman. Understanding customer expectations of service. *Sloan Management Review*, 32(3):42, 1991.

[Zhu10] Jinzy Zhu. *Cloud Computing Technologies and Applications*, pages 21 – 45. Springer US, 2010.

Appendix A

SERVQUAL

The following questionnaire has been developed by Parasuraman et al. [PZB88] to assess consumer perception of service quality. The questions labeled "E" evaluate consumer *expectation* of a service feature, while the respective questions labeled "P" evaluate consumer *perception* of the same service feature. Answers can be given on a scale from 1 - 7 (Strongly disagree - Strongly agree), where 4 represents a neutral answer and numbers 2 - 6 are unlabeled. The quality dimensions that correspond to the service features can be found in Table A.1, where Q1 represents E1 and P1, etcetera.

Quality Dimension	**Scale Items**
Tangibles	Q1. - Q4.
Reliability	Q5. - Q9.
Responsiveness	Q10. - Q13.
Assurance	Q14. - Q17.
Empathy	Q18. - Q22.

Table A.1: Relationship between SERVQUAL quality dimensions and service features [PZB88, 29].

A.1 The SERVQUAL Instrument [PZB88]

DIRECTIONS: This survey deals with your opinions of ______ services. Please show the extent to which you think firms offering ______ services should possess the features described by each statement. Do this [b]y picking one of the seven numbers next to each statement. If you strongly agree that these firms should possess a feature, circle the number 7. If you strongly disagree that these firms should possess a feature, circle 1. If your feelings are not strong, circle one of the numbers in the middle. There are no right or wrong answers — all we are interested in is a number that best shows your expectations about firms offering services.

E1. They should have up-to-date equipment.

E2. Their physical facilities should be visually appealing.

E3. Their employees should be well dressed and appear neat.

E4. The appearance of the physical facilities of these firms should be in keeping with the type of services provided.

E5. When these firms promise to do something by a certain time, they should do so.

E6. When customers have problems, these firms should be sympathetic and reassuring.

E7. These firms should be dependable.

E8. They should provide their services at the time they promise to do so.

E9. They should keep their records accurately.

E10. They shouldn't be expected to tell customers exactly when services will be performed. (—)

E11. It is not realistic for customers to expect prompt service from employees of these firms. (—)

E12. Their employees don't always have to be willing to help customers. (—)

E13. It is okay if they are too busy to respond to customer requests promptly. (—)

E14. Customers should be able to trust employees of these firms.

E15. Customers should be able to feel safe in their transactions with these firms' employees.

E16. Their employees should be polite.

E17. Their employees should get adequate support from these firms to do their jobs well.

E18. These firms should not be expected to give customers individual attention. (—)

E19. Employees of these firms cannot be expected to give customers personal attention. (—)

E20. It is unrealistic to expect employees to know what the needs of their customers are. (—)

E21. It is unrealistic to expect these firms to have their customers' best interests at heart. (—)

E22. They shouldn't be expected to have operating hours convenient to all their customers. (—)

NB.: Questions marked (—) are reverse-scored prior to data analysis.

DIRECTIONS: The following set of statements relate to your feelings about XYZ. For each statement, please show the extent to which you believe XYZ has the feature described by the statement. Once again, circling a 7 means that you strongly agree that XYZ has that feature, and circling a 1 means that you strongly disagree. You may circle any of the numbers in the middle that show how strong your feelings are. There are no right or wrong answers — all we are interested in is a number that best shows your perceptions about XYZ.

P1. XYZ has up-to-date equipment.

P2. XYZ's physical facilities are visually appeaUng.

P3. XYZ's employees are well dressed and appear neat.

P4. The appearance of the physical facilities of XYZ is in keeping with the type of services provided.

P5. When XYZ promises to do something by a certain time, it does so.

P6. When you have problems, XYZ is sympathetic and reassuring.

P7. XYZ is dependable.

P8. XYZ provides its services at the time it promises to do so.

P9. XYZ keeps its records accurately.

P10. XYZ does not tell customers exactly when services will be performed. (—)

P11. You do not receive prompt service from XYZ's employees. (—)

P12. Employees of XYZ are not always willing to help customers. (—)

P13. Employees of XYZ are too busy to respond to customer requests promptly. (—)

P14. You can trust employees of XYZ.

P15. You feel safe in your transactions with XYZ's employees.

P16. Employees of XYZ are polite.

P17. Employees get adequate support from XYZ to do their jobs well.

P18. XYZ does not give you individual attention. (—)

P19. Employees of XYZ do not give you personal attention. (—)

P20. Employees of XYZ do not know what your needs are. (—)

P21. XYZ does not have your best interests at heart. (—)

P22. XYZ does not have operating hours convenient to all their customers. (—)

NB.: Questions marked (—) are reverse-scored prior to data analysis.