Quality Management in Service Ecosystems

Christoph Riedl[1], Tilo Böhmann[1], Michael Rosemann[2] and Helmut Krcmar[1]

[1]Chair for Information Systems, Technische Universität München, Boltzmannstr. 3, 85748 Garching bei München, Germany
e-mail: {riedlc, boehmann, krcmar}@in.tum.de
[2]BPM Cluster, Faculty of Information Technology
Queensland University of Technology, 126 Margaret Street
Brisbane Qld 4000, Australia
e-mail: m.rosemann@qut.edu.au

Abstract

Service-oriented Architectures and Web services mature and have become more widely accepted and used by industry. This growing adoption increased the demands for new ways of using Web service technology. Users start re-combining and mediating other providers' services in ways that have not been anticipated by their original provider. Within organisations and cross-organisational communities, discoverable services are organised in repositories providing convenient access to adaptable end-to-end business processes. This idea is captured in the term *Service Ecosystem*.

This paper addresses the question of how quality management can be performed in such service ecosystems. Service quality management is a key challenge when services are composed of a dynamic set of heterogeneous sub-services from different service providers. This paper contributes to this important area by developing a reference model of quality management in service ecosystems. We illustrate the application of the reference model in an exploratory case study. With this case study, we show how the reference model helps to derive requirements for the implementation and support of quality management in an exemplary service ecosystem in public administration.

Keywords: Quality management, reference model, service ecosystem, quality of service, Web service, service level management, requirements engineering

1 Introduction[1]

More and more automated and semi-automated business functions are becoming available as Web services and brokers and mediators increasingly augment these services. Services are being re-combined in ways that have not been anticipated by their original providers. This phenomenon is referred to as a *Service Ecosystem* and is a new and emerging research area (Barros/Dumas/Bruza, 2005; Barros/Dumas, 2006).

Quality management is a critical management discipline whose importance on organisational success has been discussed for decades (e.g., Deming, 2000). A significant correlation between quality and profit has been shown (Gummesson, 2001). Increased quality has a positive impact on revenue, cost, and the amount of capital employed, and therefore, leads to increased productivity and higher profits. Concepts such as Total Quality Management (TQM) are widely used and actively researched (Hackman/Wageman, 1995). Since services are intangible, non-storable, and collaboratively produced their quality cannot easily be assessed prior to using the service (Fitzsimmons/Fitzsimmons, 2004). Hence, customers are required to refer to experience and to build trust with service providers.

Current research on service quality for network-based services mainly focuses on the functionality, performance, and availability of a technical service (e.g., O'Sullivan/Edmond/ter Hofstede, 2002; Canfora et al., 2004; Zeng et al., 2004). This is a severe limitation for the development of service ecosystems because this approach neglects the planning, control, and recovery of the perceived service quality from the viewpoint of service consumers. Systems narrowly focusing on technical quality cannot deliver the necessary information for consumer-focused service quality management. Thus, service quality management creates unique requirements in service ecosystems due to the high compositionality of services and sub-services as data on service quality is fundamental for both, planning and controlling services. This calls for an integrative view of quality management in service ecosystems that combines the service consumer view and the technical view on services.

The aim of this paper is to integrate the technical as well as the business aspects of quality management into a reference model to provide a comprehensive view on quality management for emerging service ecosystems. While this paper is mainly of conceptual nature, we illustrate how the reference model can be applied to derive requirements for the implementation and support of quality management by an exploratory case study in public administration.

The paper is structured as follows: In the next section, we contextualise our research by introducing the notion of service ecosystems. Next, we propose four layers of service quality management. This four layer model is then further refined into a reference model for quality management in service ecosystems. Then, an exploratory case study is presented to derive requirements for the implementation of a service ecosystems platform. Finally, we conclude the paper, discuss limitations and provide an outlook on potential pathways for future research.

2 Service Ecosystems

The increased maturity of technologies and standards facilitating the development of Web service-enabled application landscapes has made Service-oriented Architectures (SOA) one of the most popular approaches for the design of IT landscapes. The core of SOA are services designed on the premise of representing a set of tightly coupled business functions (*salesforce.com* being a prime example). Building on this, entirely new composite services that implement adaptable end-to-end business processes spanning organisational boundaries are created. This phenomenon is captured in the concept of a *Service Ecosystem*:

---

[1] **Acknowledgement:** Parts of this research have been funded by a research project within the Australian Research Council (ARC) Linkage Schema (grant code LP0669244) including financial support from SAP and the Queensland Government. This research also received funding from the German Federal Ministry of Economics and Technology (BMWi) under grant code 10MQ07024. The responsibility for the content of this publication lies with the authors.

A Web Service Ecosystem is a logical collection of Web services whose exposure and access are subject to constraints characteristic of business service delivery (Barros/Dumas, 2006).

Service ecosystems take the idea of interconnected services even further by putting constraints on the service delivery at a business level. As envisioned by Barros/Dumas (2006) in these service ecosystems, service providers of basic, or core services could augment their services by distribution and delivery functions made available to them by the ecosystem (Figure 1). For example, such an ecosystem could provide payment and metering facilities that can be used by other providers to extend the functionality of their services.

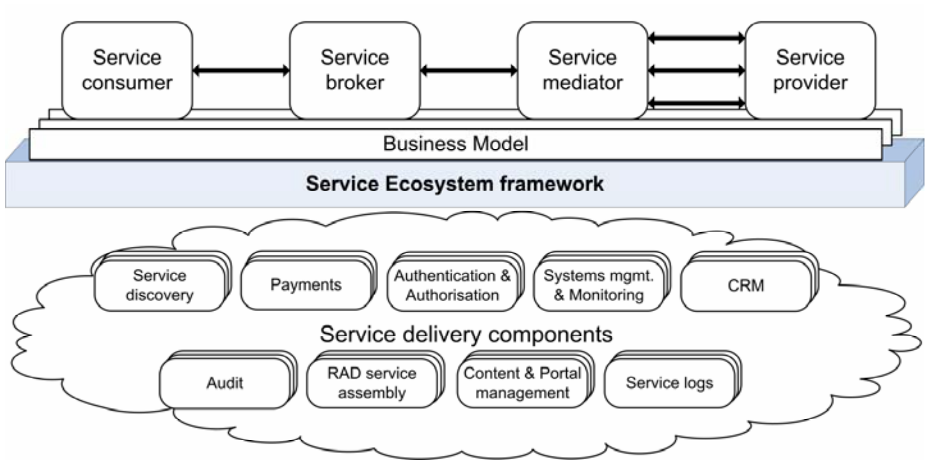

Figure 1: Top-level architecture of a Web service ecosystem (adapted from Barros/Dumas, 2006).

Service brokers bring service providers and service consumers closer together. They might also integrate a service with certain delivery functions such as payment and authentication or combine several core services into a completely new offering. Service mediators offer translations between different service formats and other routine functions to allow service brokers to concentrate on their core competencies (Barros/Dumas, 2006). Through the flow from service providers, mediators, and brokers a service is finally offered to the service consumer. This interplay of supply and distribution roles is shown in the top layer of Figure 1.

First of all, this leads to a separation of service provisioning and service delivery. A (core) service provider might no longer be responsible for the actual service delivery to a service consumer. At least the service provider might not be responsible for service delivery in every possible channel and in all service combinations as new markets and delivery channels are tapped into through reselling and brokerage. Moreover, third party delivery intermediaries (e.g. payment engines, authentication services, auction boards) augment services in ways that might not have been foreseen by the service provider. Hence, services might be delivered to new customers through new channels in unforeseen ways, which consequently leads to potentially new quality requirements.

This multi-party delivery poses critical challenges for end-to-end quality management. Service Level Management for single services is insufficient to ensure quality for service customers. Also the dynamic nature of the service composition, where Service Level Agreements are negotiated on-demand and possibly per invocation, poses challenges to quality management.

Quality management has to address this dynamic nature in two ways. First, quality relevant information has to be extracted from the system, possibly even on a per-invocation basis, to adequately evaluate service quality and address problems and incidents. Second, in the best case a service would be able to address quality problems dynamically and adapt to changes automatically.

3 Four Layers of Quality Management in Service Ecosystems

Quality is a broad topic. Based on a comprehensive literature review, we identified four main focus areas. First, marketing-driven discussions of perceived service quality by service consumers. Most notably in this field is the seminal work about SERVQUAL (Parasuraman/Zeithaml/Berry, 1988; Zeithaml/Parasuraman/Malhotra, 2002). Next are approaches that are used to define quality standards on a contractual basis – commonly referred to as Service Level Management (e.g., ITIL Service Delivery, 2003). These two address planning and controlling-oriented aspects. Moreover, there are also two operational implementation processes being researched. The first is fault management (problem and incident management) and related topics like service recovery (e.g., ITIL Service Support, 2003; Johnston/Fern, 1999). Second, a significant body of research exists related to the dynamic composition of services (e.g., Benatallah et al., 2002; Zeng et al., 2004).

This results in four interrelated layers on service quality management which are currently largely addressed independently from each other: The consumer view with its focus on perceived quality, Service Level Management, fault management, and a dynamic provisioning layer.

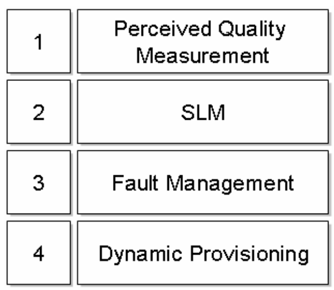

Figure 2: Four layers of quality management processes.

Service providers, however, need to consider all four layers for an integrated quality management approach. Thus, we propose the use of a four layer model that integrates existing research efforts into a consistent quality management framework for service ecosystems. The following paragraphs explain each of the proposed four layers in more detail.

Perceived Quality Measurement (Layer 1) is concerned with the measurement and evaluation of subjective quality perceptions by the consumer of any given service. This measurement is of outstanding importance in TQM (Hackman/Wageman, 1995), and, specifically with regards to services (as opposed to physical products) has been widely researched. Seminal work in this field includes the GAP Model that explains service quality as a difference score between perceived service quality and expected service quality (Parasuraman/Zeithaml/Berry, 1985). Several other service quality models have been suggested that try to explain service quality in that context (e.g., Grönroos, 1984). Yet not much research on the perception of electronic services has been conducted (Zeithaml/Parasuraman/Malhotra, 2002). Moreover, the topic of perceived quality has so far been largely ignored in the discussion of quality management in composite services and more technical research that focuses on objectively measurable attributes like availability and response time. This technical view is captured in the definition of Barros/Dumas/Bruza (2005) who address service quality management as the "extent to which non-functional properties are captured in service descriptions such that the quality of delivery can be managed in accordance to service level agreements".

On the next lower level **Service Level Management (SLM, Layer 2)** addresses the goal to maintain and improve service quality through a constant cycle of (1) agreeing, (2) monitoring, and reporting upon the achieved results in providing a service, and (3) encouraging and making suggestions for service improvement (ITIL Service Delivery, 2003). Service level targets are expressed in objective and quantitative measures of computing system availability or performance (Buco et al., 2004). Through these functions SLM aims at proactively improving the service delivery and preventing future problems through constant monitoring. However, service level agreements only address the more technical aspects of quality management and contrary to the previous layer exclusively aim at specifying objectively measurable quality targets. Optimally, SLAs are defined in such a way that they provide the right metrics to ensure sufficient quality provision. For example, complying to agreed SLAs is seen as being equivalent to high perceived service quality. However, to achieve this analogy between objective and measurable service level agreements and subjective customer perception of service quality the function of Perceived Quality Measurement is required to provide input to SLM to accurately define SLAs. Moreover, SLAs only address the next actor in a chain of services and do not focus on the final service product. The combination with Perceived Quality Measurement remedies this shortcoming since it has a wider focus and looks at the final service product.

Fault Management (Layer 3) addresses quality management in the way that it tries to fix parts of the service value chain that go wrong (through Incident Management) and to improve known problems (through Problem Management). Consequently, the focus is narrower than the one of SLM. Yet SLM provides important input as to current performance achievements (or failures to achieve the promised performance) that can be used by Fault Management to remedy those problems and improve service delivery. Fault Management is performed on top of the service operation provided by Dynamic Service Provisioning and only addresses incidents and systematic problems that cannot be solved by the underlying dynamic layer.

The lowest layer is **Dynamic Service Provisioning (Layer 4)**. This function accounts for the dynamic and on-demand aspects found in service ecosystems and is related to sub-service selection as well as performing dynamic resource allocation. This is due to the fact that Web services operate in a dynamic environment where providers might remove, modify, or relocate their services frequently and without giving prior notice (Benatallah/Sheng/Dumas, 2003). Services that have been used successfully in a previous invocation might no longer be available at a later time. Technical faults are another reason for sub-services not being available anymore. Furthermore, new services might become available that provide a *better* service and should be used instead. Thus, the selection of appropriate sub-services is typically not performed during service design but only at run-time before the actual invocation of the sub-service.

The fact that this dynamic aspect of Web services gives the opportunity to bind services dynamically at run-time is a way to enter and dismiss business relationships with a service provider on a case-by-case basis and allows optimising the delivered service quality (Keller et al., 2002). In addition, Dynamic Service Provisioning ensures that the resources for self performed services are allocated on-demand to respond to changing load and resource availability.

These four layers are linked through three feedback loops that form a cascading system consisting of three layers (Figure 3). The inner circle is a dynamic and fully automated feedback loop that chooses sub-services based on defined quality metrics and the set of available sub-services (i.e. up-and-running) that fulfil these requirements at the actual time of service invocation. This dynamic, quality-based selection of sub-services is ideally fully transparent for the service consumer. The middle circle concerns service changes and the sub-service composition. This feedback circle is mainly governed by change management that aggregates change requests from Fault Management and Service Level Management to create a coherent process for all changes concerning the service. This feedback loop captures incremental improvements to service delivery based on actual delivery experience. The outer circle focuses on redesigning the service concept based on holistic feedback from service consumers collected through Perceived Quality Measurement. These changes may trigger the redesign of the services provided to service consumers that may demand substantial changes to defined service levels and the composition of sub-services. Changes in the service concept will be generally be visible to service consumers and should lead to higher perceived quality.

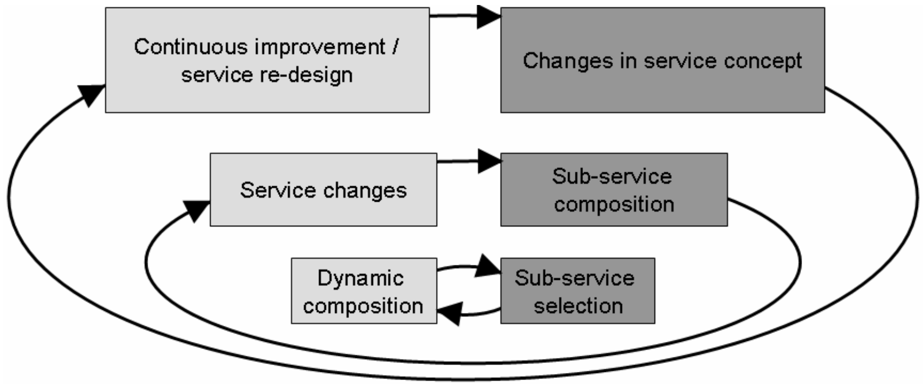

Figure 3: Three integrated feedback loops.

4 Reference Model for Quality Management in Service Ecosystems

Since quality management is an integrated management discipline that requires an unified view, we combined the four layers of quality management in service ecosystems into a reference model for quality management in service ecosystems. In the following we explain the rationale and methodology employed for the development of the reference model and present the reference model itself. Reference models serve as a conceptual framework and can be used as a blueprint for the construction of information systems (Fettke/Loos, 2003). We follow this intended use of reference models and apply it to a case study (Section 5) where it serves as a blueprint for the construction of a quality management system for service ecosystems. From this application we then derive requirements that this information system would have to meet in order to adequatly support quality management in a service ecosystem.

Good service design is crucial as it allows quality to be built into the service delivery system (Ballantyne/Christopher/Payne, 1995). Consequently, quality aspects have to be considered right from the beginning when designing a new service. Quality management across the life cycle is necessary to ensure that quality is defined, implemented, monitored, and acted upon. This concept is also found in the common Plan-Do-Check-Act cycle for continuous quality improvement (e.g., Deming, 2000). Moreover, a phased life cycle perspective facilitates process-oriented management of quality related functions within the framework. Therefore the reference model should be aligned with a life cycle model.

4.1 Comparison of Service Life Cycle Models

As motivated above a service life cycle model is needed along which the functions of the reference model can be aligned. However, there is no coherent opinion in the research community on what these life cycle phases should be and different service life cycle models have been proposed. In addition to very generic life cycle models, various specific models can be found. Those specialised models address, among others, software development (e.g., Sommerville, 2004), software deployment (e.g., Carzaniga et al., 1998), or the special requirements of composite services (e.g., Benatallah et al., 2002).To gain a more thorough understanding of possible life cycle phases, a comparison of different models found in the literature has been conducted (Table 1). The comparison includes models from various different research areas to cover the broadest possible definition of services. The last line of Table 1 contains our proposed integration model.

Source	**Phases**												
Hegering/Abeck/Neumair 1999 (Service life cycle)		Planning			Provisioning		Operating		Change				
Garschhammer et al. 2001 (Service life cycle, based on Hegering)		Design	Negotiation		Provisioning		Operation		Change			Deinstallation	
Hasselmeyer 2003 (Service life cycle)		Design			Implementation	Activation	Operation					Withdrawal	
Radisic 2002 (Service life cycle, based on Garschhammer)			Offer	Negotiation	Implementation	Installation & Test	Operation		Change			Deinstallation	
Tsalgatidou/Pilioura 2002 (Web service life cycle)		Create	Describe	Publish	Discover		Invoke					Unpublish	
Campbell/Hutchison 1994 (Service life cycle)		Conception Planning			Provision		Operation						
Sommerville 2004 (Waterfall Model)	Requirements definition	System and software design			Implementation and unit testing	Integration and system testing	Operation and maintenance						
Carzaniga et al. 1998 (Software deployment)	Release				Install	Activate			De-activate	Update	Adapt	De-install	De-release
Cox/Kreger 2005 (SOA)		Define model			Implement model	Acquire and map infrastrcture	Monitor and react						
SAP Business Process Cycle	Analyse	Design			Implement		Operate		Optimise				
SAP AG 2006	Requirements	Design			Build	Deploy	Operate		Optimise				
O'Leary 2000 (ERP Life Cycle)	Deciding to go ERP	Choosing an ERP Designing ERP			Implementing ERP		After Going Live						
Wimmer/Zerr 1995 (Service life cycle)	Decision				Integration		Use					Retirement	
Benatallah et al. 2002 (Composite service life cycle)	Wrapping native services		Service advertisement /discovery	Setting outsourcing agreements	Assembling composite services		Executing composite	Monitoring		Evolving			
zur Muehlen 2004 (Process life cycle)	Goals, environmental analysis, organisational analysis	Process design			Process implementation		Process enactment	Monitoring		Process evaluation			
Proposed integration	**Analysis & Design**		**Negotiaton**		**Provisioning**		**Use**					**Withdrawal**	

Table 1: Comparison of service life cycle models.

Following this comparison, an integrated life cycle model has been developed (Figure 4). Contrary to most life cycle models present in the comparison the following paragraphs explicate each phase, justify our selection and grouping of phases and illustrate the activities covered in each phase.

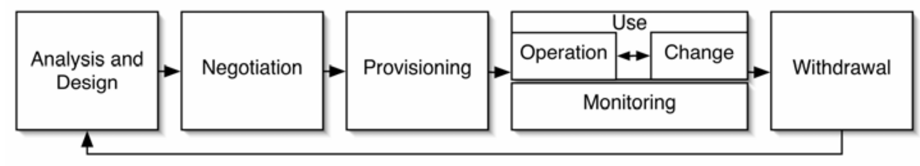

Figure 4: Integrated service life cycle model.

Analysis and Design An analysis phase is only mentioned by the SAP Business Process Cycle. However, most other models assume business goals or objectives as an input (i.e., requirements) for their design phase. A design phase is mentioned in almost all models (except for Radisic (2002), Carzaniga et al. (1998) and Benatallah et al. (2002)) and has hence been included in the integrated model. To limit the total number of phases in our model and to avoid unnecessary complexity, analysis and design have been combined.

Negotiation Although an explicit negotiation phase is only mentioned by Garschhammer et al. (2001), the combined activities mentioned by the other models justify the use of a separate phase. For our model we call this phase negotiation. This phase covers more than the actual contract negotiation taken into account by Garschhammer et al. as it also covers the process of describing, offering, publishing, and advertising of a service.

Provisioning Depending on the field of application of the model, the next phase can include very different activities. For simplicity this phase has been called provisioning and includes all activities necessary to prepare the actual operation of a service. Depending on the type of service and the degree of vertical integration this might include a whole software or system development life cycle or software deployment life cycle. This phase also covers testing of the service.

Use An operation phase is again mentioned by most of the models. We studied some of the models include an explicit change or optimisation phase after the operation phase that would ensure continuous development and improvement. Hence our model includes an use phase comprising an operation and a change function that are linked by constant monitoring.

Withdrawal The life cycle is completed, when the service becomes obsolete or for other reasons is no longer provided. Although different names (de-installation, unpublish, or retirement) are used, a similar phase can be found in most models. In this phase everything necessary to remove the service (logically or physically) is performed.

4.2 Process Architecture Framework

As proposed by Becker et al. (2002) the architecture framework of our reference model uses the reference design “house” suggested by Meise (2001) as a basis. In the next step, the integrated service life cycle model from the comparison developed earlier has been added to the framework in order to group and order the processes. Then the four quality management layers have been added (Perceived Quality Measurement, Service Level Management, Fault Management, and Dynamic Service Provisioning).

Looking back at the life cycle phases shows that the functions performed during the design phase and the withdrawal phase have not yet been covered as they are not part of the four layers of quality management. Hence, the function groups Analysis and Design, and Termination have been added to the framework. The Termination function covers the activities that have to be performed in the withdrawal phase. All life cycle activities are therewith covered in the framework.

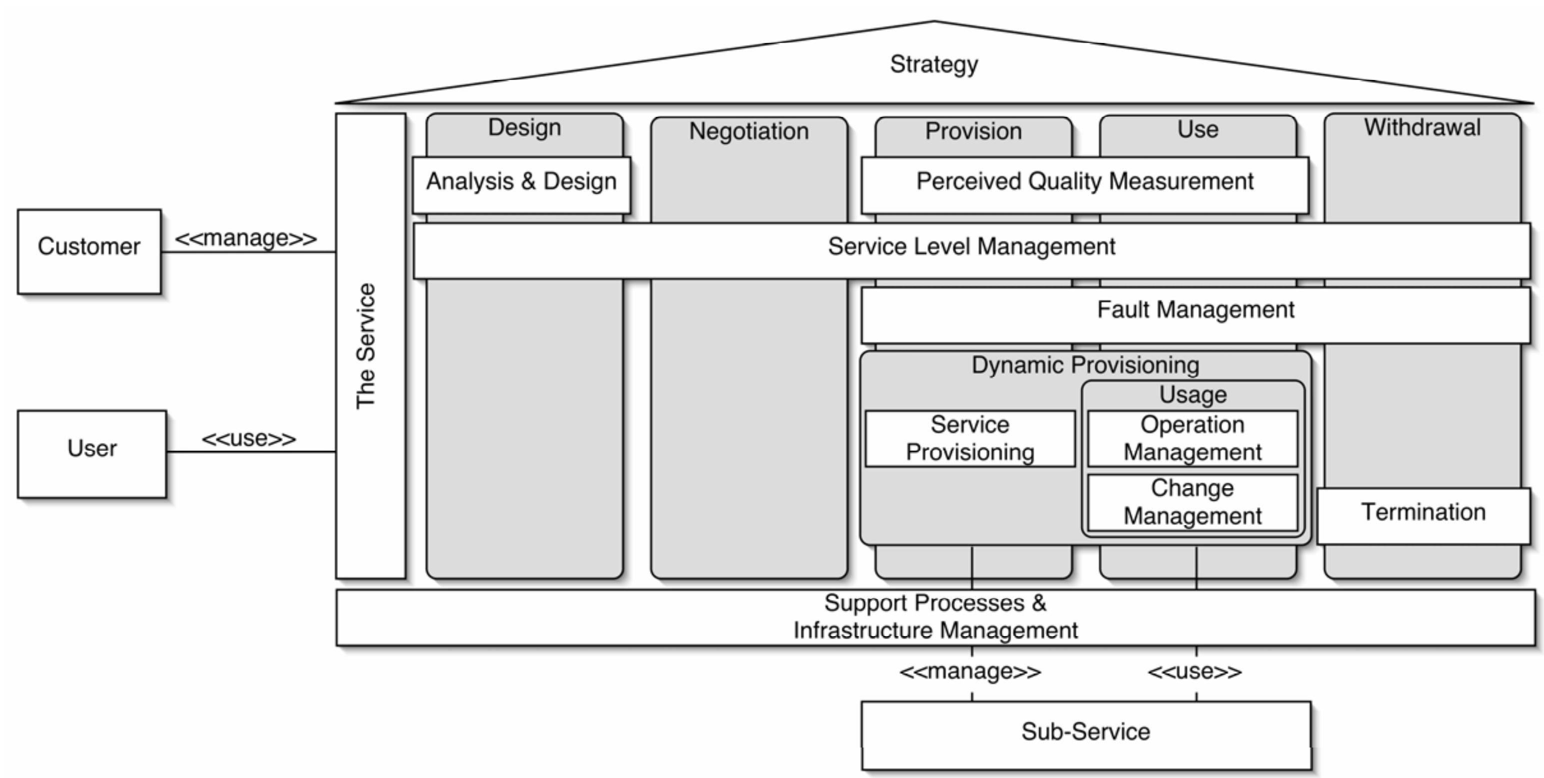

Figure 5: Full process architecture framework.

ITIL functions comprising the activities performed during Dynamic Service Provisioning have been added to the framework for more details. However, they remain grouped under the Dynamic Service Provisioning function. In this step the function Service Provisioning,Operation Management, and Change Management are introduced.

Furthermore, the model integrates a recursive service model that breaks the consumer role up into an user and a customer role. This model can then be applied recursively for services that rely on other sub-services to provide their service (cf. Garschhammer et al., 2001 and Rodosek, 2003). The resulting complete process architecture framework is shown in Figure **Fehler! Verweisquelle konnte nicht gefunden werden.**.

The following sections give a brief outline of the activities that have to be performed within each of the functions and describe a process to show what quality related information they use, in what form the information is required, what quality information is produced, and how it is used and transformed.

4.3 Perceived Quality Measurement

As motivated in the four layers of quality management this function addresses the quality management from the perspective of the service consumer. This area of the reference model consists of various functions.

Plan Quality Measurement The function includes defining what service quality is, identifies its underlying dimensions, and determines how it can be conceptualised and measured (Zeithaml/Parasuraman/Malhotra, 2002). Moreover, service providers have to identify who the actual consumers of their services are. Through the use of brokers in a service ecosystem, a certain service might be delivered through a channel not originally anticipated by the provider of this service.

Measure Service Quality This function comprises the actual measurement of service quality as defined in the previous step. Measurement data has to be made available both in raw measurements as well as in aggregated form. The methods for measuring perceived quality from a customer perspective can be classified as event-based, attribute-based, and problem-based (Bruhn, 2006). The following table provides links to sample methods that can be used to perform the quality measurement.

Type of measurement	*Method*
Event-based	Critical Incident Technique (Flanagan, 1954) Sequential Event Method (e.g., Lovelock/Wright, 1999)
Attribute-based	Multi-Attribute Measuring Methods (e.g., Zeithaml/Parasuraman/Malhotra, 2002) Decompositional Methods such as conjoint analysis (e.g. Churchill/Iacobucci, 2004)
Problem-Oriented	Problem-Detecting-Method (Bruhn, 2006) Complaint Analysis (Benkenstein/Holtz, 2001)

Table 2: Types of quality measurement.

Evaluate and Improve After service quality has been measured, any negative quality perceptions by the customer have to be transformed into service improvements. However, services in a service ecosystem are highly dynamic and can include a complete hierarchy of sub-services, thus linking customer’s perceived quality (or lack thereof) to the individual sub-service (combination) that was the source of this quality perception is critical. Very detailed quality information is necessary to identify what customer dissatisfaction is, for example, it might be linked to a failing payment sub-service rather than to the actual service being provided. Moreover, since sub-services might be bound only at run-time, individual service invocations might use completely different sub-services. Thus, to handle complaints and to analyse quality judgments by customers the exact service compositions that were actually used need to be stored. Another aspect is that service brokers might choose to deliver a core service through a different channel not originally foreseen by the core service provider. Thus, a service providing adequate quality in one channel might not perform well in another.

In case of any gaps between expected and actual service quality, the service has to be improved. This can be done in three steps following the three integrated feedback loops introduced earlier (see Figure 3).

1. For external services, the selection criteria have to be adapted to reflect customer's quality perception (dynamic composition loop).
2. Those parts of the service that are provided internally have to be improved. This activity is closely linked with Operations Management, the maintenance function of Service Level Management, and Change Management (service changes loop).
3. The general service concept, including the service decomposition has to be improved. This activity is closely linked to Service Analysis and Design (continuous improvement loop).

Perceived Quality Measurement targets the service consumer. Hence, it can only be performed during those phases of a service life-cycle where the service is operational and actively sought by consumers. Consequently, the Perceived Quality Measurement function spans the production phases which are the provision and use phase in our reference model. Here it allows to derive service improvements from running operations.

4.4 Service Level Management

Service Level Management (SLM) covers the functions of defining, negotiating, monitoring, reporting, and controlling service agreements that have been signed with customers. The idea is to set and formalise performance goals, which are then monitored in terms of service levels. The goal of such a function is to maintain and improve service quality through a constant cycle of (1) agreeing, (2) monitoring and reporting upon the achieved results in providing a service, and (3) encouraging and making suggestions for service improvement (ITIL Service Delivery, 2003). Service level targets are expressed in objective and quantitative measurements of computing system availability, performance, business process efficiency, or effectiveness (Buco et al., 2004).

The main input to SLM is the service design as it is the basis for defining measurable service properties, which in turn can then be monitored and evaluated. In addition to performance reports and service evaluations the monitoring data produced by SLM is used directly or via forecasting algorithms to predict changes in performance for forecasting and capacity planning purposes (Dan et al., 2004). Furthermore, it is used for billing purposes as it contains information about actual resource usage and violations (that might be linked to penalties). Finally, monitoring data is enquired by Fault Management for finding causes of problems.

ITIL defines SLM as the central process for the definition, negotiation and control of service quality for IT services. This underscores that SLM starts with the design of a service, supports the negotiation of the desired performance of a service, and monitors a service according to agreed metrics throughout their life cycle. SLM thus integrates quality management for services in all phases from design to termination.

4.5 Fault Management

The Fault Management process in this framework is divided into Incident Management, Problem Management, and Service Recovery processes.

Incident Management Incidents are deviations from the expected standard operation of a system or service (ITIL Service Support, 2003). The goal of Incident Management is to restore normal service operation, i.e. service operation within the agreed service level limits as quickly as possible by whatever means necessary.

Problem Management Problems are conditions that have been defined, identified from a larger incident, or several incidents exhibiting common symptoms for which the cause is not known (ITIL Service Support, 2003). The goal of Problem Management is to minimise the impact that incidents and problems have on normal service operation and to ensure stable operation. In order to achieve this goal, Problem Management has both reactive and proactive aspects.

Service Recovery Contrary to the best intents to provide quality service, occasional failures are inevitable. Once a failure has occurred, the service provider must be prepared to recover from this failure in order to resume operation and try to restore customer satisfaction (Johnston/Fern, 1999; Shaw/Craighead, 2003). Service recovery and its influences on customer satisfaction and retention have been widely studied in service management research (e.g., Bell/Zemke, 1987; Bitner/Booms/Tetreault, 1990; Johnston/Fern, 1999).

The dynamic and on-demand character of a service ecosystem poses specific problems on Fault Management due to the demand for run-time service compositions. In the case that a service execution fails due to a sub-service failure, the exact matching of which specific sub-service has been used for this particular service invocation has to be made available (possibly in real-time) for Incident and Problem Management. Offering complete information on every single service invocation puts unique demands on the information logistics for a quality management information system.

Faults can only occur during the operational phases of the service life-cycle but not during the design or negotiation phase. Only while a service is running errors can occur. Hence, Fault Management spans all phases except design and negotiation in our framework.

4.6 Dynamic Service Provisioning

To account for the highly dynamic and on-demand aspects found in service ecosystems the functions Service Provisioning, Operation Management, and Change Management have been grouped together to handle these dynamic aspects. A service ecosystem is dynamic in two ways. First, the selection of sub-services is performed on-demand. Second, the resources for self-performed services are allocated on demand to respond to changing load and resources availability. The input to control both processes are the service design, customer SLAs and their evaluation, as well as business objectives (Dan et al., 2004).

As the name suggests Dynamic Provisioning takes place during the provision and use phase of a service when the actual service product is produced. This is reflected in our reference model by spanning those two phases.

4.7 Supporting and Related Processes

The core functions from our four layers of quality management for service ecosystems are accompanied by two supporting functions: Service Analysis and Design and Service Termination. Moreover, the function Dynamic Service Provisioning is comprised of the functions Service Provisioning, Operation Management, and Change Management. This section highlights the relationships they have to the other areas to understand the complete information flow in the overall reference model.

Service Analysis and Design Service analysis and design is concerned with the process of developing new or the adaptation of existing services. This function receives feedback from Fault Management, Perceived Service Quality Measurement, and Service Termination as inputs to create new or improve existing service designs. These service designs are in turn used by Service Level Management, Service Provisioning, and Operation Management.

Service Provisioning Service Provisioning is entered after a service agreement has been reached with a customer and is concerned with all processes necessary to properly install the agreed service. The inputs to this function are the service description provided by Analysis and Design as well as the SLAs provided by Service Level Management.

Operation Management Operation Management comprises, as defined by ITIL, all activities and measures necessary to enable and/or maintain the intended use of ICT (Information and Communications Technology) services and infrastructure in order to meet Service Level Agreements and business targets (ITIL ICT, 2003). Operation Management receives input from SLAs from Service Level Management and in turn provides alerts to Fault Management and tuning measures to Change Management.

Change Management Change Management ensures the use of standardised methods and procedures to handle all changes to the production environment in order to keep the impact of change-related problems on service quality to a minimum. Thus, all other functions that wish to modify a service or the underlying infrastructure in any way interface with Change Management.

Service Termination The service life cycle comes to an end with the termination of the service. This function, however, can provide valuable feedback for Analysis and Design that can be used to adapt existing services or design new ones.

5 Case Study

To demonstrate the use of the reference model and to further illustrate the idea behind our four-layered approach we present a descriptive case study. Using the case study we try to illustrate how the functions of the reference model can be applied to derive requirements for service quality management. Moreover, from the application of our reference model to a case study several challenges could be identified. From these challenges that we encountered during the application of the case study we derived requirements that a service ecosystem platform would have to address in order to fully support the quality management approach proposed in our reference model.

The case study has been drawn from a research collaboration with one of the departments of Queensland (QLD) State Government located in Brisbane, Australia in the time from January to March 2007. During this time, government agencies were discussing a number of services that could be implemented within the service ecosystem of the state government. Government agencies developed conceptual designs for these services. Such a conceptual design provides sufficient information to anticipate challenges for quality management during the lifetime of the service and thus for demonstrating the application of the reference model, irrespective of an actual future implementation of the service. The service explored in our case study is the *Business Name Renewal* service targeted at Queensland public and business owners who would access the services via the Smart Service Queensland (SSQ) portal Web site by Queensland Government under *www.qld.gov.au*. The service would allow business owners to renew their business name registrations online for the payment of a small fee.

The portal through which the service would be offered is outside the jurisdiction of the department we conducted the case study with and is part of a larger service ecosystem in which various agencies offer raw services to a broader public. The portal provider plays the role of a service broker that delivers a service through a particular channel. The department plays the role of a service provider that offers a service in the ecosystem. The payment service plays the role of a service delivery component that is used by other service providers to create a marketable service. Figure 7 gives an abstract outline of the government service ecosystem as it is currently implemented and used by other services already offered by Queensland Government.

5.1 Layer 1: Perceived Quality Measurement

Plan Quality Measurement: The service provided by the state government would only be invoked by the SSQ portal, which acts as a service broker. The only channel that this broker delivers the service to is the *www.qld.gov.au* portal Web site. Consequently, the only service consumers are the users of this portal site. Next, the quality criteria on which these service consumers base their subjective service quality judgment have to be identified and actual service perception has to be gathered. Although the service is currently only delivered through the *www.qld.gov.au* portal Web site this might change in the future. Australia Post, for example, which is already offering several government services (e.g., passport applications), could decide to offer the Business Name Renewal service as well. Moreover, delivery via various channels for mobile devices could be added.

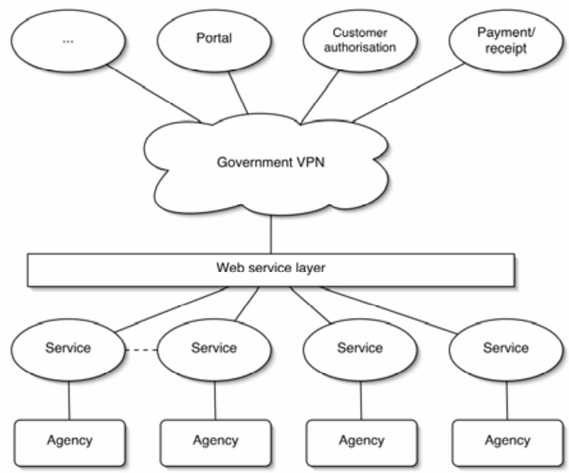

Figure 6: Service ecosystem of Queensland Government.

Measure Service Quality: To measure service quality the Multi-Attribute Measuring Method could be used by conducting a survey among the identified service consumers. A modified version of the survey developed by Fassnacht/Koese (2006) could be used for that purpose, covering issues such as graphic quality, clarity of layout, ease of use, and reliability.

Evaluate and Improve: Since the overall Business Name Renewal service delivered to service consumers would involve three services (the Business Name Renewal service itself, the SSQ portal for service delivery through a Web site, and a payment processing sub-service), measurements have to be evaluated accordingly. If problems emerge, it has to be determined if users are not satisfied with the Business Name Renewal service itself, the way the information is presented through the SSQ portal, or maybe with the way payments are processed and what payment methods are accepted. Consequently, those parts of the overall service that need to be improved have to be identified.

Thus, holistic high-level quality information about the service consumers' perception of the overall service quality can be collected using a survey approach. This information is then evaluated to derive quality improvements for the complete service.

Requirements: The following table summarises the requirements from this part of the case study.

Case Characteristics	*Derived Platform Requirement*
Different groups of service consumers	Platform support to easily identify service consumer to facilitate perceived service quality evaluations with the actual service user.
Currently single-channel, potentially multiple channels in the future	Platform support to easily identify the channels through which a service is delivered for channel specific quality evaluation.
Dynamic and varying ways of service usage and service combinations	Platform support for easy identification of final service products to facilitate holistic, perceived quality judgments.

Multiple views on service quality	Platform support for service evaluation based on subjective quality perception. E.g., through the use of user ratings, comments, or recommendation mechanisms.
Quality measurement of composite services	Platform support to aggregate and disaggregate perceived quality measurements to support improvements of composite services as well as individual service components.
Identification of life cycle phase	Platform support to identify which life-cycle phase a service is in. Only then is it possible to support the different quality management activities in the different phases.

Table 3: Summary of requirements for Perceived Quality Measurement.

5.2 Layer 2: Service Level Management

Since the service is targeted at the general public no customer SLAs would be in place and no customer negotiation would take place. However, a SLA can been defined by the business with internal service provisioning on behalf of the anonymous service consumers of the service. That way, objectively measurable quality targets are set against which service operation can be compared to assess quality degradations. Moreover, a SLA would be in place between the SSQ portal and the deparment offering the Business Name Renewal service and another one between the portal and the payment processing sub-service.

For the internal SLA several Quality of Service (QoS) parameters like *response time* and *availability* are defined. In addition to a detailed specification of the QoS parameter itself the point of measurement, the data source, an aggregate model, a measurement schedule, and respective thresholds for the parameter are defined. With this QoS data in place, SLM can take up operation by monitoring the QoS in the specified way and alerting crossed thresholds and coordinating maintenance operations. Moreover, SLM can help to detect future developments (e.g. in increased server load) that allow proactive management.

Using input from the Perceived Quality Measurement layer above, the development of QoS parameters can be guided to ensure that meeting SLAs is linked to high quality, as the actual service users perceive it. Thus, the survey items used in Perceived Quality Measurement regarding the technical quality can be broken down and translated into response time thresholds of the Business Name Renewal service that, together with the additional response times by the portal and the payment sub-service would result in positive consumer evaluation.

After the QoS parameters have been defined, the actual monitoring of the service components and the underlying computing infrastructure begins. The collected data is checked against the specified thresholds to identify quality problems during service delivery. If a certain threshold is crossed, alarms are raised and a technician tries to solve the problem.

During the operation of such a service it would thus be possible to detect that the computing infrastructure operating the Business Name Renewal service reaches its capacity limit due to increased demand. SLM proposes to add more resources to keep up with the increased demand.

This section showed how a complete cycle of agreeing, monitoring, and suggestions for service improvements as proposed by the reference model has been completed.

Requirements: The following table presents requirements we identified that are relevant for SLM.

Case Characteristics	*Derived Platform Requirement*
SLAs with external partner	Platform support for flexible and reliable negotiation, monitoring, and performance improvements with external partners.
SLA with internal provisioning	Platform support for real-time monitoring of internally

	provided service parts to facilitate adequate resource allocation and management to accommodate changing demand and service outages.
Multi-party services	A governance structure is required that establishes who has the interest, authority, and resources to evaluate the quality of a composite service.

Table 4: Summary of requirements for Service Level Management.

5.3 Layer 3: Fault Management

Some service invocations suddenly start failing. The portal provider collects various operational data through the server logs and an integrated customer feedback form. The service portal operates normally (other services report no problems) so they forward the problem reports to the department offering the Business Name Renewal service. Comparing the time-stamps in the problem reports it can be discoverd that the failing service invocations used the newly emerged payment provider Z. Using this information our department augments the sub-service selection criteria model to exclude recently failing providers. After this change in the sub-service selection model the service operates normally. Since the state government organisation would be interested in using the cheaper provider Z (rather than one of the alternative services offered) but is reluctant to do so due to the service's unreliability they forward the problem reports to provider Z to help them offer the service in satisfying quality. However, the departmant decides to always use payment provider Z as a recovery service if the otherwise preferred service becomes unavailable.

This interplay of Incident Management, Problem Management, and Service Recovery shows how, with the use of appropriate monitoring data, incidents can be resolved. More importantly it also shows how service quality can be maintained or maybe even improved through a feedback cycle within Fault Management.

Requirements: A summary of requirements derived from the application of Fault Management in the case study is shown in the table below.

Case Characteristics	*Derived Platform Requirement*
Composite services	Tracking support of complete service invocations that allows pinpointing problems to individual sub-services or components.
Problem tracking	Platform support to monitor individual service invocations as sub-services are selected per-invocation which complicates problem management.
Distributed service recovery	Service recovery strategies need to be supported that allow augmenting a composite service in real-time according to actual sub-service availability.

Table 5: Summary of requirements for Fault Management.

5.4 Layer 4: Dynamic Service Provisioning

As the previous paragraph demonstrated Dynamic Service Provisioning queried the global service repository to discover another payment service that could be used instead. Moreover, if one of the providers becomes unavailable an alternative service with similar functionality can be used and, based on a predefined sub-service selection model, the "best" available sub-services can be selected. Thus, the quality of the provided service can not only be maintained but also sometimes even increased due to clever sub-service selection.

The second aspect of Dynamic Service Provisioning is the dynamic resource allocation. To meet the defined response time thresholds, sufficient resources have to be available to deliver the Business Name Renewal service. The monitoring data gathered by SLM is used to detect an increased demand and more resources can be added. This might happen especially if new services are offered by the department and provisioned through the same systems. In that case, more resources might be necessary to keep service delivery performance within the defined thresholds.

Requirements: Again, a table summarises the requirements from the case study.

Case Characteristics	*Derived Platform Requirement*
Dynamic sub-service selection	Platform support for the dynamic discovery, binding, and usage of sub-services.
Dynamic adoption to changing situations	Platform support for dynamic adaptation of selection criteria for sub-services to allow dynamic adoption to changing user perceptions.
Changing demand of service	Platform support for dynamic resource allocation to accommodate changes in service demand.

Table 6: Summary of requirements for Dynamic Service Provisioning.

5.4 Case Study Summary

The case study shows how the use of our four layer approach enables an integrated quality management that identifies service consumers and their quality perception, uses this information to define relevant SLAs, supports a comprehensive fault management, and finally addresses the dynamic delivery of the services.

However, several challenges were encountered while trying to apply the reference model to the case study. From these challenges several requirements for necessary features of a service ecosystem platform were derived.

6 Conclusions

The paper has shown how the complex and dynamic aspects of service ecosystems pose special requirements on quality management that are insufficiently addressed by existing approaches. Through the separation of service delivery from service supply, services are delivered to unknown service consumers, possibly through a wide selection of channels. Here quality is very important as it is directly linked to revenue. Through dynamic sub-service selection, providers have to ensure that their services are selected over those offered by competitors. This can only be achieved through exceptional quality that constantly satisfies both, customers (e.g., a broker) and service consumers (e.g., the user of a travel-booking website).

Taking the learnings from the case study, concrete requirements for the implementation of a service ecosystem platform have been derived. These requirements are summarised in the tables above.

Addressing quality management in the highly dynamic context of service ecosystem is a challenging research topic. Four layers of quality management for service ecosystems together with the reference model and the developed platform requirements are a first step in addressing this topic. However, empirical validation would be necessary to show if and how the proposed functions and processes influence service quality.

References

Ballantyne, D./Christopher, M./Payne, A. (1995) Improving the Quality of Services Marketing: Service (Re)design is the Critical Link. Journal of Marketing Management, 11(1-3):7–24

Barros, A./Dumas, M. (2006) The Rise of Web Service Ecosystems. IT Professional, 8(5):31–37

Barros, A./Dumas, M./Bruza, P. (2005) The Move to Web Service Ecosystems. BPTrends, November, 1–9

Becker, J et al. (2002) Konfigurative Referenzmodellierung. In Becker, J./Knackstedt, R. (eds.): Wissensmanagement mit Referenzmodellen. Physica-Verlag, Heidelberg

Bell, C. R./Zemke, R. E. (1987) Service Breakdown: The Road to Recovery. Management Review, 76(10):32–36

Benatallah, B. et al. (2002) Declarative composition and peer-to-peer provisioning of dynamic Web services. In Proceedings of 18th International Conference on Data Engineering (ICDE'02). IEEE Computer Society, Los Alamitos, CA, 297–308

Benatallah, B./Sheng, Q.Z./Dumas, M. (2003) The Self-Serv environment for Web services composition. IEEE Internet Computing, 7(1):40–48

Benkenstein, M.; Holtz, M. (2001) Qualitätsmanagement von Dienstleistungen. In: Bruhn, M.; Meffert, H., editors, Handbuch Dienstleistungsmanagement, 193–209. Gabler, Wiesbaden, 2nd edition

Bitner, M. J./Booms, B. H./Tetreault, M. S. (1990) The Service Encounter: Diagnosing Favorable and Unfavorable Incidents. Journal of Marketing, 54(1):71–84

Buco, M. J. et al. (2004) Utility computing – SLA management based upon business objectives. IBM Systems Journal, 43(1):159– 178

Bruhn, M. (2006) Qualitätsmanagement für Dienstleistungen. 6. Auflage. Springer Berlin Heidelberg

Campbell, A./Coulson, G./Hutchison, D. (1994) A quality of service architecture. SIGCOMM Computer Communication Review, 24(2):6–27

Canfora, G. et al. (2004) A Lightweight Approach for QoS-Aware Service Composition. In Proceedings of 2nd International Conference on Service Oriented Computing (ICSOC'04)

Carzaniga, A. et al. (1998) A Characterization Framework for Software Deployment Technologies. Technical Report CU-CS-857-98, Dept. of Computer Science, University of Colorado

Cox, D. E./Kreger, H. (2005) Management of the service-oriented-architecture life cycle. IBM Systems Journal, 44(4):709–726

Churchill, G. A.; Iacobucci, D. (2004) Marketing Research: Methodological Foundations. Thomson Learning, Mason, OH, 9th edition

Dan, A. et al. (2004) Web services on demand: WSLA-driven automated management. IBM Systems Journal, 43(1):136–158

Deming, W.E. (2000) Out of the Crisis. MIT Press, Cambridge, Massachusetts

Fassnacht, M. /Koese, I. (2006) Quality of Electronic Services – Conceptualizing and Testing a Hierarchical Model. Journal of Service Research, 9(1):19–37

Fettke, P./Loos, P. (2003) Classification of reference models: a methodology and its application. Information Systems and e-Business Management (ISeB), 1(1)35-53

Fitzsimmons, J.A./Fitzsimmons, M.J. (2004) Service management: operations, strategy, and information technology. McGraw-Hill, New York, NY

Flanagan, J. C. (1954) The Critical Incident Technique. In: Psychological Bulletin, 51(4):327-358

Garschhammer, M. et al. (2001) Towards Generic Service Management Concepts – A Service Model Based Approach. In 7th IEEE/IFIP International Symposium on Integrated Network Management (IM 2001). IEEE Computer Society, 719–732

Grönroos, C. (1984) A Service Quality Model and its Marketing Implications. European Journal of Marketing, 18(4):36– 45

Gummesson, E. (2001) Productivity, Quality and Relationship Marketing in Service Operations. In: Bruhn, Manfred/Meffert, Heribert (eds.): Handbuch Dienstleistungsmanagement. 2nd edition. Gabler, Wiesbaden, 851–872

Hackman, J.R./Wageman, R. (1995) Total Quality Management: Empirical, Conceptual, and Practical Issues. Administrative Science Quarterly, 40(2):309–342

Hanemann, A./Schmitz, D./Sailer, M. (2005) A Framework for Failure Impact Analysis and Recovery with Respect to Service Level Agreements. In Proceedings of the 2005 IEEE International Conference on Services Computing (SCC '05). Washington, DC: IEEE Computer Society, 49–58

Hasselmeyer, P. (2003) An infrastructure for the management of dynamic service networks. Communications Magazine, IEEE, 41(4):120–126

Hegering, H-G/Abeck, S./Neumair, B. (1999) Integrated Management of Networked Systems. Morgan Kaufmann Publishers, San Francisco, CA

ITIL ICT (2003) ITIL ICT Infrastructure Management. Crown, electronic edition

ITIL Service Delivery (2003) ITIL Service Delivery. Crown, electronic edition

ITIL Service Support (2003) ITIL Service Support. Crown, electronic edition

Johnston, R./Fern, A. (1999) Service Recovery Strategies for Single and Double Deviation Scenarios. The Service Industries Journal, 19(2):69–82

Keller, A. et al. (2002) Managing dynamic services: A contract based approach to a conceptual architecture. In Network Operations and Management Symposium (NOMS 2002). IEEE Publishing, 513– 528

Meise, V. (2001) Ordnungsrahmen zur prozessorientierten Organisationsgestaltung. Verlag Dr. Kovac, Hamburg

O'Leary, D. E. (2000) Enterprise Resource Planning Systems. Cambridge University Press, Cambridge, UK

O'Sullivan, J./Edmond, D./ter Hofstede, A. (2002) What's in a Service? Distributed and Parallel Databases, 12:117– 133

Parasuraman, A./Zeithaml, V. A./Berry, L. L. (1985) A Conceptual Model of Service Quality and Its Implications for Future Research. Journal of Marketing, 49(1):41–50

Parasuraman, A., Zeithaml, Valerie A./Berry, L. L. (1988) SERVQUAL: A Multiple-Item Scale for Measuring Consumer Perceptions of Service Quality. Journal of Retailing, 64(1):12–40

Radisic, I. (2002) Using Policy-Based Concepts to Provide Service Oriented Accounting Management. In Network Operations and Management Symposium (NOMS 2002). IEEE Publishing, 313–326

Rodosek, G. D. (2003) A generic model for IT services and service management. In IFIP/IEEE 8th International Symposium on Integrated Network Management, IEEE Publishing, 171– 184

SAP AG, (2006) IT Support Organization Solution Map – Edition 2006. In: http://www.sap.com/solutions/businessmaps/, accessed 2007-11-26

SAPSolutionMap SAP AG Business Process Cycle Overview. https://www.sdn.sap.com/irj/sdn/bpx-cycle, accessed 2007-11-26

Shaw, N. G./Craighead, C. W. (2003) Technology Enablers to Recover from Failures in E-Services. Communications of the ACM, 46(6):56–57

Sommerville, I. (2004) Software Engineering. 7th edition. Pearson/Addison-Wesley, Boston London

Sturm, R.; Morris, Q.; Jander, M. (2000) Foundations of Service Level Management.

SAMS, Indianapolis

Tsalgatidou, A./Pilioura, T. (2002) An Overview of Standards and Related Technology in Web Services. Distributed and Parallel Databases, 12(2):135–162

Wimmer, F./Zerr, K. (1995) Service für Systeme – Service mit Systemen. Absatzwirtschaft, 38(7):82-87. Citet after: Kleinaltenkamp A, Begriffsabgrenzung und Erscheinungsform von Dienstleistungen, pages 28–50. In: Handbuch Dienstleistungsmanagement. Gabler, Wiesbaden, 2nd edition

Zeithaml, V. A./Parasuraman, A./Malhotra, A. (2002) Service Quality Delivery through Web Sites: A Critical Review of Extant Knowledge. Journal of the Academy of Marketing Science, 30(4):362–375

Zeng, Liangzhao et al. (2004) QoS-Aware Middleware for Web Services Composition. IEEE Transactions on Software Engineering, 30(5):311–327

zur Mühlen, M. (2004) Workflow-based Process Controlling. Logos Verlag, Berlin