# Variety of QoS - the MNM Service Model Applied to Web Hosting Services

*Martin Sailer*
*Munich Network Management Team*
*University of Munich*
*Oettingenstr. 67*
*D-80538 Munich*
*sailer@nm.ifi.lmu.de*

*Andreas Hanemann, David Schmitz*
*Munich Network Management Team*
*Leibniz Supercomputing Center*
*Barer Str. 21*
*D-80333 Munich*
$\{$*hanemann*$|$*schmitz*$\}$*@lrz.de*

**Abstract**

In today's IT environments customers often have to rely on best-effort services such as IP and e-mail. In order to have a competitive advantage, service providers offer different kinds of guarantees for their services which are often called *Quality of Service (QoS)* parameters. Parameters in this sense are service-specific, and are not limited to classical network QoS parameters (e.g. delay, jitter, loss). However, customers and providers still share no common understanding regarding the specification of these parameters.

To address this issue we propose a QoS specification process composed of two steps: model the service in question and specify the QoS parameters with respect to the resulting model. In this paper, we identify requirements for an extension of the MNM Service Model in order to support the mentioned specification process. Therefore, we analyze a scenario, the Web Hosting Service offered by the Leibniz Supercomputing Center, to identify typical QoS aspects. We then use the example scenario to gain requirements for an extension of the model for service-oriented QoS parameters.

**Keywords**

Service-oriented QoS, QoS specification, QoS modeling

## 1 Introduction

The guarantees which are offered by providers for their IT services to their customers are often called *Quality of Service* parameters. With the paradigm shift towards service orientation, the perception of QoS has changed from mainly technical criteria on the network layer to a set of parameters that describe quality aspects of a service in a customer-oriented way. However, the specification of these parameters is subject to many ambiguities. For instance, different points of measurement result in deviating measured values, even if the parameters bear the same name. This imposes new requirements on QoS specification, such as expressing the parameters in a way, both customers and providers understand them. Given the variety of QoS parameters, an appropriate abstraction becomes necessary in order to establish a common understanding of how service-oriented QoS parameters are specified.

To address this issue we propose a QoS specification process composed of two key steps: model the service in question and specify the QoS parameters with respect to the resulting model. By using an abstraction based on a service model, QoS parameters can be specified in a declarative way and become thus more apparent to customers.

For that reason, we identify requirements for extending the MNM Service Model [6], a generic service management model, for QoS. We chose that model, because it defines a common terminology in generic service management, specifies atomic roles and denotes the major building blocks a service is composed of. To perform the requirement analysis we first apply the MNM Service Model to a example scenario in order to retrieve and structure different aspects of QoS. Afterwards, we assess general principles concerning the extension of the model.

The plan of this paper is as follows. The example scenario, the Web hosting service of the Leibniz Supercomputing Center, is described in Section 2. A short overview of the MNM Service Model is given in Section 3. We proceed by presenting a systematic analysis of requirements towards an extension of the model in Section 4. The last section concludes the paper and presents future work.

## 2 QoS Aspects for the Web Hosting Scenario

We have selected the Web Hosting Service of the Leibniz Supercomputing Center (LRZ), because this service is appropriate to show many kinds of different QoS aspects. The LRZ, which is the computing center for the Munich universities runs the scientific network in Munich and hosts web sites for other research institutions. Figure 1 shows the resources which are used for the provisioning of the Web Hosting Service. For load sharing and redundancy reasons the service is provided on five different web servers. The load sharing is performed by a level 4/7 switch. Static web sites are located in a database using the Andrew File System, while dynamic web pages are located in a database run by the Network File System. The LRZ's own web sites are located on four redundant servers. The E-Mail Service of the LRZ can also be accessed by using a dedicated web mail server. In case of severe network problems inside the LRZ an emergency server is available which only contains some web sites containing basic information.

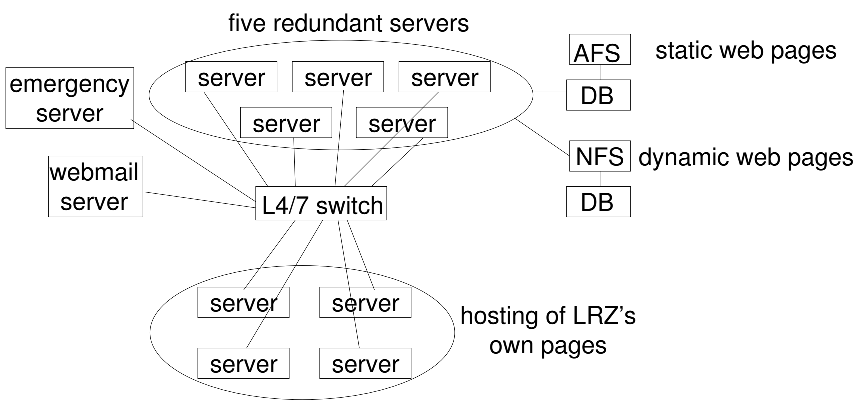


**Figure 1** Web Hosting Service at the LRZ

We identified several QoS parameters in this scenario regarding not only functional but also management-related aspects. We begin with a description of QoS parameters that address the functionality of the Web Hosting Service.

*Functional QoS parameters of the Web Hosting Service* The following parameters have been derived from operational parameters as agreed upon the LRZ and its customers, e.g. the University of Munich. A similar set of parameters can also be found in [15] and [10].

- **availability**
  Usually expressed by the term downtime and specified as a percentage of time (e.g. hours per month): the duration of service unavailability due to failure
- **reliability**
  This indicates the number of errors per total number of requests and responses generated or received by the service.
- **delay**
  The delay that occurs when accessing a website, usually specified in milliseconds.
- **document accesses**
  This specifies the total number of simultaneous requests that can be handled by the Web Hosting Service.

It is important to note, that the specification of these parameters is subject to certain ambiguities: for instance, the service quality can be perceived from different viewpoints (e.g. service user, service provider) and on different points of measurement (e.g. service access point, provider-internal). The actual QoS measurement can also be carried out in various ways. While some approaches rely on passive monitoring of user interactions, others actively test the offered features. Additionally, it is sometimes the aim to monitor the quality a user receives (e.g. delay when requesting a web site), while in other cases a test user performs typical requests and its perceived QoS is mapped onto the real users (synthetic transactions). There are also different possibilities how to perform the comparison between target values and actual values of the QoS parameters (e.g. different granularity, sample rates).

Beside these general parameters, there also exist more specific parameters for this service. For example, the content of an institution's web site could be quickly accessible, but it is outdated (see [13] for a more detailed discussion of this problem). If this is caused by the local browser cache, then the LRZ has no influence on it and this will not be part of the agreement between the LRZ and the institute. But to reduce the network traffic, there are also caches at the server side which could cause the provisioning of outdated content. The operational parameters of this cache (e.g. maximum delay, percentage of outdated pages) could be part of the agreement.

Note finally, that we are not referring to quality as it is often done by web hosting companies to differentiate their service from services of their competitors. Quality in this sense would be the available disk space, installed software on the hosting machine (PHP, Perl, MySQL, etc.), or the granting of a root access. These quality features should not be called QoS parameters, because they are static and therefore no online monitoring is necessary. Accordingly, these aspects are modeled as part of the service functionality in the MNM Service Model (see Section 4).

*Management-related QoS parameters of the Web Hosting Service* Addressing the functionality of a service leads to a plethora of quality relevant parameters, but it is only one aspect of QoS. The management interface between service provider and customer, usually subsumed under the term Customer Service Management (CSM), enables customers to individually monitor and control their subscribed service. It forms an integral part of a service, and needs thus to be considered when dealing with Quality of Service. Examples of such QoS parameters for the university institute are the time the LRZ server can be reached to perform content changes, and the bandwidth available to transfer the updated pages.

Besides the use of its own resources the Web Hosting Service also depends on other services like the Domain Name Service and the IP Service (connectivity). If the user accesses the hosted web sites via the Virtual Private Network Service there is also a dependency from this service. It is obvious that the QoS a user perceives for the Web Hosting Service depends on the QoS provided by its subservices. If there is a failure in the routing tables or the DNS is not available, the availability of the Web Hosting Service is also affected. The same is true if there is a high delay for the response of DNS request. This can lead to high delays for web site access and also to situations where a user sees outdated content caused by caching.

As these examples show, many different aspects of QoS exist and there is a need for structuring these aspects. We proceed now with a short overview on the MNM Service Model, which serves as a basis for the requirement analysis presented in Section 4.

## 3 MNM Service Model

The MNM Service Model [6], which was developed by the Munich Network Management Team, is a generic model for IT service management. It distinguishes between *customer side* and *provider side*. The customer side contains the basic roles *customer* and *user*, while the provider side contains the role *provider*. The provider makes the service available to the customer side. The service as a whole is divided into usage which is accessed by the role user and management which is used by the role customer. The explicit modeling of the service management on the right hand side of the model in contrast to service usage on the left hand side is an important feature of the model.

The model consists of two main views. The *Service View* (see Fig. 2) shows a common perspective of the service for customer and provider. Everything that is only important for the realization of the service is not contained in this view. For these details another perspective, the *Realization View*, is defined (see Fig. 3).

The Service View contains the *service* for which the functionality is defined for management as well as for usage. There are two access points (*service access point* and *Customer Service Management* access point) where user and customer can access the usage and management functionality, respectively. The *Customer Service Management* (CSM) [14] is a concept of a single management interface between customer and provider where all management issues like the ordering of new services, reporting function, information about service troubles can be exchanged. E.g. during the service ordering the negotiation of the QoS parameters is performed using the CSM. Associated to each service

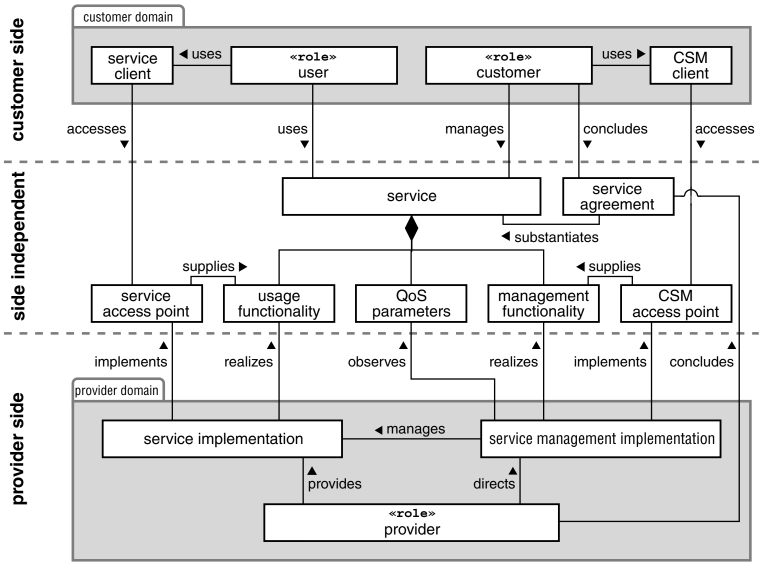


**Figure 2** Service View of the MNM Service Model

is a list of QoS parameters which have to be met by the service at the service access point. The QoS surveillance is performed by the *service management implementation*.

In the Realization View the *service implementation* and the *service management implementation* are described in detail. For both there are provider-internal resources and subservices. For the *service implementation* a *service logic* uses internal resources (devices, knowledge, staff) and external subservices to provide the service. Analogous, the service management implementation includes a *service management logic* using *basic management functionalities* [11] and external management subservices.

The MNM Service Model can be used for a similar modeling of the used subservices, that is, the model can be applied recursively.

## 4 Requirements for Extending the MNM Service Model for QoS

The main purpose of modeling with the MNM Service Model is to establish a common understanding between customers and providers about the semantics of QoS parameters in order to support the QoS specification process. Therefore, the MNM Service Model should be able to deal with all important aspects of QoS parameters.

On the one hand, with the current MNM Service Model it's already possible to define one important aspect of QoS modeling: the reference point where the measurement of parameters takes place. The MNM Service Model contains classes like *client*, *service access point* or *service logic* with which can be started to specify where the acquisition of parameters is performed.

But on the other hand, the MNM Service Model contains only one not nearer specified class for the QoS parameters itself, for which no attributes or modeling requirements

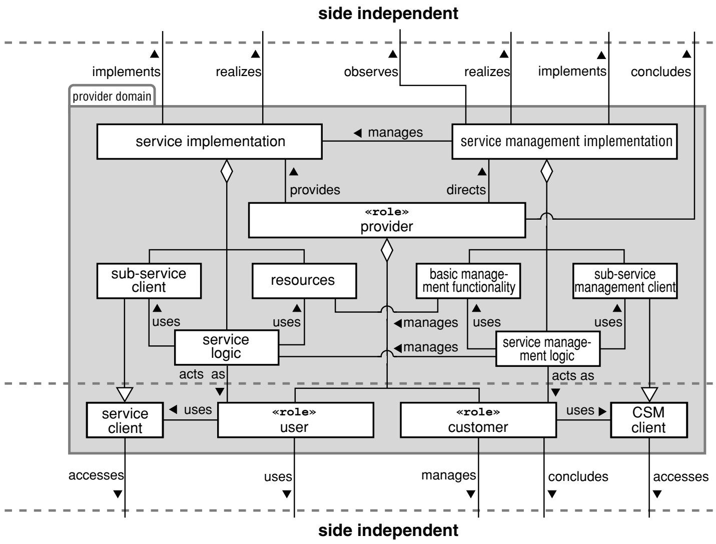


**Figure 3** Realization View of the MNM Service Model

have been defined. As a result, many aspects of QoS cannot be expressed explicitly in the current model. For example there is no possibility to define the semantics of a QoS parameter, the measurement methodology or how the actual values are correlated with the desired values. Therefore, it is necessary to identify all important aspects of QoS modeling and extend the MNM Service Model to support them explicitly.

After having described QoS aspects of the Web hosting Scenario (Section 2) as well as the MNM Service Model (Section 3), we now perform a more systematic analysis of requirements towards an extension of the model. Therefore, our requirement analysis process is composed of two key steps: analyze the identified QoS aspects with respect to the MNM Service Model and assess general principles.

According to the sketched pattern, we first apply the Service Model to the Web Hosting Scenario by following the methodology presented in [7]. Based on the resulting instantiation of the model, we then identify reference points, to which the described QoS aspects (see Section 2) apply. As we focus on *where can QoS aspects be attached to*, this approach can be seen as bottom-up. In the second step, we concentrate on general requirements that need to be met when extending the MNM Service Model. Thus, we address issues arising from the question *how should the extension be carried out* in a top-down manner.

## 4.1 Identification of Reference Points in the Instantiation of the Service Model

Section 2 denoted the variety of QoS aspects arising in a Web Hosting service. Due to limited space in this paper not all combinations of aspects for a QoS parameter could be covered in the example. Nevertheless, the example revealed different aspects which can help to classify QoS parameters. Mainly, these aspects differ according to their point of measurement, the viewpoint they are perceived from and how measurement is actually performed. Based on this classification, we identified two reference points in the Service Model Instance of the Web Hosting Service (depicted in Figure 4). As already stated in Section 2 we understand Customer Service Management as an integral part of the service, and consider therefore parts of the model concerned with CSM in our analysis as well.

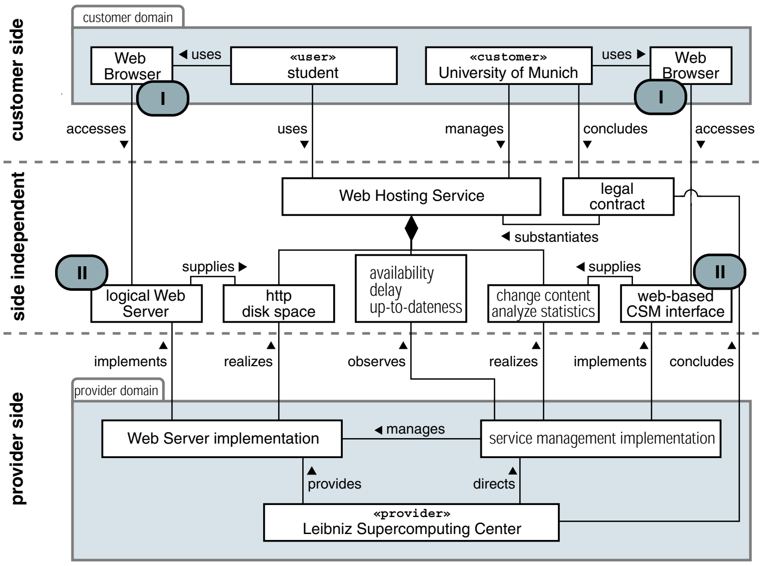


**Figure 4** Service Model Instance of the Web Hosting Service

I When a user requests a web site, he perceives a certain type of QoS, often referred to as end-to-end QoS. Accordingly, the relation between the *service client* (browser) and the *service access point* (logical web server) is relevant for the extension of the model. In a similar manner, QoS perceived by the *customer* (University of Munich) when accessing the web-based CSM interface needs to be considered. Since measurement of end-to-end QoS in that sense is carried out in the customer's domain, it is of great importance, that an unambiguous QoS specification between provider and customer is in place. The specification has to entail, whether the measurement is performed for each user individually, or is done using transactions made by synthetic users. Also, characteristics of the user's service client influence the measurement process, and should therefore be considered.

II The Service Access Point, which resides in the side independent part of the model, can be used for implementation independent QoS measurement. For this reason, invocations of basic primitives for accessing the logical web server (request, response) are intercepted and correlated [8]. In that manner, the response time between corresponding requests and responses of the server can be calculated. As further logical components performing the interception are required, the model has to be extended accordingly. Again, the mentioned QoS aspects also apply to the management part of the model.

Besides the denoted aspects, which largely focus on the customer side and the side independent part, requirements can also be derived from the provider side of the model. As the details of the provider side are contained in the Realization View we present two further reference points with respect to the implementation view of the Web Hosting Service (depicted in Figure 5).

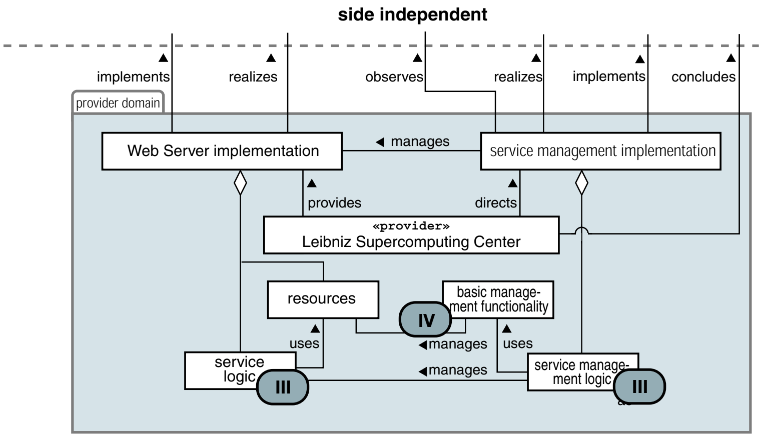


**Figure 5** Implementation Model Instance of the Web Hosting Service

III QoS parameters can also be gathered from the application service logic itself by injecting code (e.g. by implementing a QoS measurement methodology using ARM [1]). By that means, the total number of simultaneous requests or the time interval between http-requests and http-responses can be obtained from the execution environment of the web server.

IV In today's provisioning of IT services the QoS a user perceives is in most cases not measured at the service access point, but an aggregation of the information gained by the network and system management is used to estimate the QoS.
In the scenario, the basic management functionality performs an aggregation of resource parameters (e.g. web server logs, process time, server utilization, available disk space), which are used as threshold parameters. Based on these parameters monitoring of the service is performed and used to detect possible SLA violations.
As a result, the mapping of service-oriented QoS parameters onto the parameters of resources has to be covered. Dreo [2] proposed a modeling of the resource parameters as Quality of Device (QoD) parameters. She also approached the problem of mapping the QoS parameters onto the QoD parameters.

### 4.2 General principles for extending the model

Until now, the MNM Service Model contains a class QoS parameters, which is capable of serving as a starting point for a QoS extension of the model. The latter can be achieved by generalizing the identified QoS aspects and mapping them into the Service Model. For that, we are now going to identify general principles, which we have derived from the methodology that has led to the model itself [6] as well as the survey of Jin and Nahrstedt [12]. Accordingly, a further improvement of the model has to consider the following requirements:

**Genericity:** The QoS definition should be applicable to all kinds of services (as generic as the whole model).

**Abstraction level:** The abstraction level of the QoS extension has to match to the previous MNM Service Model.

**Expressiveness:** The QoS definition should be as declarative as possible so that it can be read by a human reader (customer-centric). On the other hand the definition has to be precise enough to avoid future ambiguities. Both aims at achieving a better understanding between customer and provider.

**Semantics of QoS parameters:** The model should contain a possibility to describe the semantics of a QoS parameter.

**Measurement methodology:** The model should support a specification of the measurement methodology: this should include for example if the measurement is active or passive or if synthetic transactions are used or not.

**Relationship of of actual and desired values:** Firstly, it should be possible to specify the desired value ranges for QoS parameters. Secondly, a specification should be included to describe what has to be done (concerning the customer) if the actual values differ (e.g. SLA penalties, escalation intervals, reports).

**QoS mapping:** In many other models a gap can be found between the high-level user interface and the low-level service realization. In the QoS extension the mapping of abstract QoS parameters onto the features of devices (also called Quality of Device parameters [3]) should be possible.

**Implementation independent:** Like the current MNM Model the QoS extension should be independent from any implementation. The reason of this is to make it possible for the customer to easily compare the offers of different providers for QoS parameters. The need to adapt to this kind of modeling helps to draw a clear border between the provider-dependent service implementation and the service access point which usage should be independent from the actual implementation.

**Service life cycle:** The QoS definition should be applicable to all phases of the service life cycle. While most known QoS definitions deal only with the usage phase of a service, it should also be possible to define QoS parameters for other phases. For example, the negotiation between customer and provider could contain certain time intervals which define the maximum duration until a customer request for a service has to be answered.

## 5 Conclusion and Future Work

In this paper we analyzed requirements for an extension of the MNM Service Model for QoS aspects. Therefore we applied the MNM Service Model to a example scenario in order to retrieve and structure different aspects of QoS. Based on the resulting instantiation of the model we identified reference points to which these QoS aspects apply. Additionally, we assessed general principles concerning the extension of the model.

The main idea behind the extension of the model is to support the QoS specification process taking place between customers and providers. By modeling the service in question with the extended MNM Service Model, QoS parameters can be specified with respect to the model and become thus more apparent to customers.

Further directions in our work include the definition of a extension of the Service Models which fulfills the requirements identified in this paper. Our second focus is to further extend the model for service-oriented event correlation[9]. This kind of event correlation differs from the event correlation that current network and system management tools offer as it integrates customer reports about service problems into the correlation process. With the correlation of these service events it is possible to map problems with services onto malfunctioning resources. This modeling allows to improve the mean time to repair (MTTR) which is often part of service level agreements between customer and provider.

### Acknowledgment

The authors wish to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on previous versions of this paper. The MNM Team directed by Prof. Dr. Heinz-Gerd Hegering is a group of researchers of the University of Munich, the Munich University of Technology, and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. Its web–server is located at http://wwwmnmteam.informatik.uni-muenchen.de.

## References

[1] Application Response Measurement (ARM) API . Technical Standard C807, ACM Transactions on Graphics, July 1998.

[2] G. Dreo Rodosek. *A Framework for IT Service Management*. Habilitation, Ludwig-Maximilians-Universität München, June 2002.

[3] G. Dreo Rodosek. A Generic Model for IT Services and Service Management. In *Integrated Network Management VIII — Managing It All*, volume 2003, pages 171–184, Colorado Springs, USA, March 2003. IFIP/IEEE, Kluwer Academic Publishers.

[4] Patricia Florissi. *QuAL: Quality Assurance Language*. PhD thesis, Columbia University, Department of Computer Science, New York, NY 10027, 1995.

[5] Svend Frolund and Jari Koistinen. QML: A Language for Quality of Service Specification. Technical Report HPL-98-10, Hewlett-Packard Laboratories, Palo Alto, 1998.

[6] M. Garschhammer, R. Hauck, H.-G. Hegering, B. Kempter, M. Langer, M. Nerb, I. Radisic, H. Roelle, and H. Schmidt. Towards generic Service Management Concepts – A Service Model Based Approach. In G. Pavlou, N. Anerousis, and A. Liotta, editors, *Proceedings of the 7th International IFIP/IEEE Symposium on Integrated Management (IM 2001)*, Seattle, Washington, USA, May 2001. IEEE Publishing.

[7] M. Garschhammer, R. Hauck, H.-G. Hegering, B. Kempter, I. Radisic, H. Roelle, and H. Schmidt. A Case–Driven Methodology for Applying the MNM Service Model. In R. Stadler and M. Ulema, editors, *Proceedings of the 8th International IFIP/IEEE Network Operations and Management Symposium (NOMS 2002)*, pages 697–710, Florence, Italy, April 2002. IFIP/IEEE, IEEE Publishing.

[8] Markus Garschhammer. *Formale Dienstgütespezifikation zur Steuerung ihrer rechnergestützen Umsetzung im Dienstlebenszyklus – in German*. PhD thesis, University of Munich, Department of Computer Science, Germany, to appear in 2004.

[9] A. Hanemann and D. Schmitz. Service-Oriented Event Correlation - Workflow and Information Modeling Approached. In *Proceedings of the 3rd International Workshop on Distributed Event-Based Systems (DEBS 2004)*, Edinburgh, Scotland, May 2004.

[10] H. Hazewinkel, C. Kalbfleisch, and J. Schoenwaelder. RFC 2594: Definitions of Managed Objects for WWW Services. Technical report, May 1999.

[11] H.-G. Hegering, S. Abeck, and B. Neumair. *Integrated Management of Networked Systems – Concepts, Architectures and their Operational Application*. Morgan Kaufmann Publishers, ISBN 1-55860-571-1, 1999.

[12] Jingwen Jin and Klara Nahrstedt. QoS Specification Languages for Distributed Multimedia Applications: A Survey and Taxonomy. In *IEEE Multimedia Magazine*, to appear 2004.

[13] Alexandros Labrinidis and Nick Roussopoulos. Update propagation strategies for improving the quality of data on the web. In *The VLDB Journal*, pages 391–400, 2001.

[14] M. Langer, S. Loidl, and M. Nerb. Customer Service Management: A More Transparent View To Your Subscribed Services. In A. S. Sethi, editor, *Proceedings of the 9th IFIP/IEEE International Workshop on Distributed Systems: Operations & Management (DSOM 98)*, Newark, DE, USA, October 1998.

[15] Jong-Tae Park and Jong-Wook Baek. Web-based Intranet/Internet Service Management with QoS Support. *IEICE Transactions on Communications*, E82-B(11), November 1999.