



Nr.: FIN-005-2009

Auswahl von Web Services auf Basis von Qualität mit  
Hilfe der Multi-Attributiven Nutzentheorie

Bastian Grabski, Lars Krüger, Steffen Zimmerhäkel

*Arbeitsgruppe Wirtschaftsinformatik*



Fakultät für Informatik  
Otto-von-Guericke-Universität Magdeburg

# Technical Report

Nr.: FIN-005-2009

**Auswahl von Web Services auf Basis von Qualität mit  
Hilfe der Multi-Attributiven Nutzentheorie**

Bastian Grabski, Lars Krüger, Steffen Zimmerhäkel

*Arbeitsgruppe Wirtschaftsinformatik*



Fakultät für Informatik  
Otto-von-Guericke-Universität Magdeburg

**Impressum** (§ 5 TMG):

*Herausgeber:*

Otto-von-Guericke-Universität Magdeburg  
Fakultät für Informatik  
Der Dekan

*Verantwortlich für diese Ausgabe:*

Otto-von-Guericke-Universität Magdeburg  
Fakultät für Informatik  
Lars Krüger  
Postfach 4120  
39016 Magdeburg  
E-Mail: lars.krueger@ovgu.de

<http://www.cs.uni-magdeburg.de/Preprints.html>

*Auflage:* 60

*Redaktionsschluss:* März 2009

*Herstellung:* Dezernat Allgemeine Angelegenheiten,  
Sachgebiet Reproduktion

*Bezug:* Universitätsbibliothek/Hochschulschriften- und  
Tauschstelle

# Auswahl von Web Services auf Basis von Qualität mit Hilfe der Multi-Attributiven Nutzentheorie

Bastian Grabski, Lars Krüger, Steffen Zimmerhäkel

AG Wirtschaftsinformatik  
Otto-von-Guericke-Universität Magdeburg  
Postfach 4120  
39016 Magdeburg  
{bgrabski | lkrueger}@ovgu.de  
steffen.zimmerhaekel@accenture.com

**Abstract:** Web Services sind die populärste Möglichkeit, um Services in einer Service-Oriented Architecture (SOA) zu realisieren. Trotz aller positiven Prognosen konnte SOA noch keinen großen Erfolg vorweisen. Ein Grund ist das Finden und die Auswahl von geeigneten Web Services. Derzeit mangelt es immer noch an geeigneten Web-Service-Spezifikationsstandards, in denen die Qualität berücksichtigt wird. Mit diesem Artikel soll ein Vorgehensmodell für die Auswahl von Web Services vorgestellt und anschließend gezeigt werden, wie die Multi-Attributive Nutzentheorie genutzt werden kann, um ausgewählte Web Services zu bewerten und anschließend zu priorisieren.

## 1 Einführung

Web Services sind die aktuell populärste Möglichkeit zur Umsetzung der Services innerhalb einer Service-orientierten Architektur (SOA). Doch trotz aller Erfolgsprognosen der letzten Jahre, konnte sie diese bisher nicht erfüllen und bleibt hinter den Erwartungen zurück [DGH03]. Dafür gibt es verschiedene Gründe. Ein schwerwiegender Grund ist, dass die Möglichkeiten zur Spezifikation, welche die geltenden Web Services Standards eröffnen, nicht ausreichend sind. Qualitätseigenschaften, die von erheblicher Relevanz für die Auswahl und Komposition von Web Services seitens des Nutzers sind, werden dabei unzureichend berücksichtigt [OT05]. Gäbe es einen freien Markt für Web Services, könnten Web Services mit gleichen oder ähnlichen Funktionalitäten parallel von mehreren Anbietern angeboten werden. Die angebotenen Web Services können sich jedoch in ihrer Qualität unterscheiden. Selbst bei nur einem Anbieter wäre eine Differenzierung verschiedener Qualitätsstufen in Abhängigkeit vom Preis denkbar [VHA05].

Ein Nutzer wird einen Web Service aus der Menge aller angebotenen Web Services gleicher Funktionalität in der Regel auf Grund dessen qualitativer Eigenschaften, wie z. B. Latenz oder Verfügbarkeit wählen. Daraus ergeben sich folgende Fragestellungen. Wie kann ein Nutzer die Qualitätseigenschaften von Web Services bestimmen und wie kann er die daraus gewonnenen Informationen in seinem Sinne nutzen? Auf welche Art und Weise ist es möglich, aus einer Vielzahl verschiedener Merkmale Kriterien mit in den Auswahlprozess einzubeziehen?

Das vorliegende Paper gibt hierzu einen Überblick und zeigt etwaige Problematiken auf. In Kapitel 2 werden sowohl Grundlagen vorgestellt, die Standards und Qualität von Web Services umfassen, sowie auch auftretende Probleme innerhalb dieser Bereiche umrissen. Im folgenden Kapitel 3 wird ein generisches Vorgehensmodell zur Web Service Auswahl vorgestellt. Diese wird in den Zusammenhang mit der Multi-Attributiven Nutzentheorie gestellt. Abschließend folgt eine Zusammenfassung in Kapitel 4.

## 2 Grundlagen

### 2.1 Web Services

Ein Web Service ist eine Softwarekomponente, die eine Menge von Diensten anderen Softwarekomponenten in einem Netzwerk zur Verfügung stellt, und die diese über ein wohldefiniertes Interface unter Verwendung von XML-basierten Nachrichten nutzen können [DGH03], [KT02], [KST02]. Bei den grundlegenden Standards zu Web Services handelt es sich im Wesentlichen um die Web Service Description Language (WSDL), das Simple Object Access Protocol, auch Service Oriented Architecture Protocol (SOAP<sup>1</sup>) und Universal Description, Discovery, and Integration (UDDI). Diese Standards basieren wiederum auf XML und benötigen ein Transportprotokoll (z. B. HTTP) [Ce02], [W3C07a].

Weitergehende Informationen zu qualitätsspezifischen Kriterien lassen sich mit der WSDL nicht formulieren. Es gibt semantische Erweiterungen der WSDL, die von dem W3C lediglich den Status einer Empfehlung erhalten haben. Sie sind somit keine geltenden Standards [W3C07b].

Für die Kommunikation zwischen Web Services werden XML-Nachrichten auf Grundlage von SOAP genutzt. Es stellt ein standardisiertes Rahmenwerk zum Transport von beliebigen Informationen zur Verfügung ohne Berücksichtigung der Semantik des Inhalts [ACKM04].

Im UDDI existiert die allgemeine Datenstruktur *tModel*. Mit ihrer Hilfe können Anbieter Aspekte des Web Service beschreiben. Sie wird jedoch als nicht ausreichend für die Spezifikation von Web Services und die Darstellung von qualitätsrelevanten Informationen betrachtet [Do07], [OT05].

---

<sup>1</sup> Seit der Version 1.2 vom Juni 2003 wird SOAP als eigenständiger Name benutzt und nicht länger als Akronym [W3C07a].

## 2.2 Web Service Discovery

Um die Relevanz einer vollständigen Spezifikation zu verdeutlichen, soll an dieser Stelle der Discovery Prozess nach dem W3C vorgestellt werden. Unter Web Service Discovery wird die Suche nach einer maschinenlesbaren Beschreibung eines Web Services, unter Berücksichtigung funktionaler und nicht-funktionaler Kriterien verstanden. Die folgenden Beschreibungen orientieren sich, soweit nicht anders gekennzeichnet, ausschließlich an den Ausführungen des W3C [W3C07a].

Der in Abbildung 1 dargestellte Discovery Prozess kann sowohl automatisiert, als auch manuell durchgeführt werden:

1. Der Discovery Service (z. B. UDDI-Register) erhält eine Web Service Beschreibung (WSB) und eine dazugehörige funktionale Beschreibung (FB) vom Provider. Bei der funktionalen Beschreibung kann es sich nur um einige Metadaten, z. B. den Uniform Resource Identifier (URI), aber auch um komplexere Formate wie das tModel (im UDDI) oder eine Sammlung von Semantiken oder Ontologien (z. B. RDF, DAML-S, OWL-S) handeln, die in einer maschinenlesbaren Form vorliegen.
2. Der nachfragende Nutzer übermittelt seine Anforderungen an den Discovery Service, welcher die Auswahlkriterien mit denen der verfügbaren FB vergleicht.
3. Der Discovery Service sendet WSB von Web Services, welche die Anforderungen des Nutzers erfüllen, an diesen. Daraus wählt die nachfragende Entität den passenden Web Service.

Anschließend kann nun der gewählte Web Service aufgerufen werden. Auch dies kann grundsätzlich manuell oder automatisiert umgesetzt werden.

Das W3C betont ausdrücklich, dass die Architektur des Discovery Prozesses kein Augenmerk darauf richtet, wie der Discovery Service zu den Spezifikationen der Web Services kommt. Dies differiert je nach Art und Weise der Implementierung des Discovery Service. Ist dieser in Form einer Suchmaschine realisiert, so kann er das Internet nach vorhandenen Beschreibungen durchsuchen und diese sammeln. Bei einem UDDI dagegen müssten die Web Service Provider die Daten in dem Register veröffentlichen.

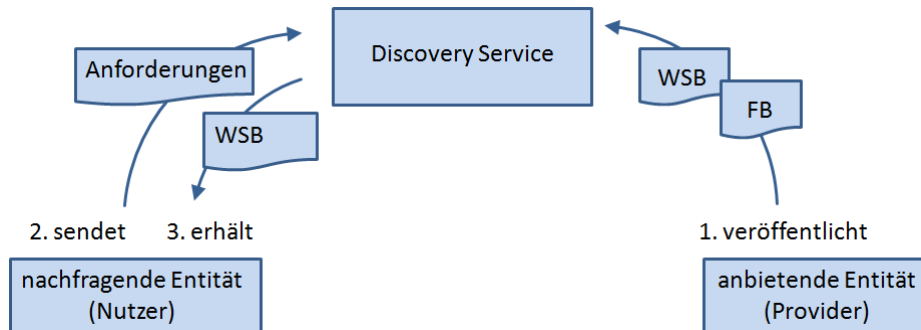


Abbildung 1: Discovery Prozess nach [W3C07a].

Für die Realisierung des Discovery Service nennt das W3C drei Möglichkeiten:

- Register (UDDI),
- Index und
- Peer-to-Peer-System.

Festzuhalten ist, dass ein Anbieter von Web Services Informationen über diese an geeigneter Stelle, in geeignetem Format und in ausreichender Dichte bereitstellen muss, damit sie von potentiellen Nutzern gefunden und gewählt werden können. Für einen Nutzer ist nicht von vornherein klar, welche Informationen über einen Web Service verfügbar sind und wie diese codiert sind.

Für die Spezifikation von Web Services und die Realisierung des Discovery Service existiert kein einheitlicher Standard. Dadurch ergeben sich Probleme bei der Suche nach geeigneten Web Services, da sowohl verschiedene Spezifikationsmöglichkeiten, als auch verschiedene Arten von Discovery Services existieren. In den letzten Jahren wurde viel geforscht zu diesem Thema und etliche Ansätze sind entstanden, die von Erweiterungen des UDDI [Ra03], über verbindliche, erweiterte Spezifikationsvorschläge [OT05], eigenen Spezifikations Sprachen [TPPE05], bis hin zu individuellen Umsetzungen des Discovery Service reichen [AM07a], [AM07b], [BHMS05]. Jedoch hat sich bis heute nichts als de facto Standard etablieren können.

## 2.3 Qualität

Wie eingangs erwähnt, sind die Qualitätseigenschaften von Web Services aus Nutzersicht von besonderem Interesse bei der Suche und Auswahl von Web Services. Nachfolgend wird definiert was Qualität ist und in welchem Zusammenhang sie mit Web Services verwendet wird.

Die DIN EN ISO 9000:2005 definiert Qualität als den Grad, in dem ein Satz inhärenter Merkmale Anforderungen seitens bestimmter Interessengruppen erfüllt [CEN05]. Inhärente Merkmale werden verstanden als der Sache innewohnende Kennzeichen. Sie ermöglicht unterschiedliche Perspektiven auf den Begriff Qualität, die von der Zugehörigkeit zur jeweiligen Interessengruppe abhängen. So wird der Hersteller eines Produktes unter Umständen andere Anforderungen an sein Produkt stellen als der Kunde. Wenn im Weiteren von Qualität verwendet wird, so wird darunter die Qualität aus Nutzersicht verstanden.

Nach Garvin gibt es fünf Betrachtungsweisen von Qualität mit verschiedenen Ansätzen. Dies sind der transzendente, der produktbezogene, der fertigungsbezogene, der wertbezogene und der nutzerbezogene Ansatz [Ga84]. Da dieser Arbeit eine nutzerorientierte Betrachtungsweise zugrunde liegt, wird auch nur der diesbezügliche Ansatz näher vorgestellt. Demnach steht der Nutzer im Mittelpunkt der Betrachtung. Qualität wird dadurch bestimmt, inwieweit ein Produkt in der Lage ist, die Bedürfnisse des Nutzers zu befriedigen und zur Nutzerzufriedenheit beizutragen [Ba98]. Neben dieser fitness-for-use Sichtweise, bedeutet Qualität auch die Abwesenheit von Defiziten und Fehlern [Ju98].

Eine Qualitätseigenschaft ist eine Eigenschaft, die zur Abgrenzung von Sachen in qualitativer oder quantitativer Hinsicht verwendet werden kann [Ba98].

## **2.4 Quality of Service**

Wer in der Literatur etwas über Qualität im Zusammenhang mit Web Services sucht, stößt unweigerlich auf den Begriff Quality of Service (QoS). Im Deutschen wird dafür oftmals der Begriff Dienstgüte gebraucht [Ru06]. Viele Methoden zur Suche und Komposition von Web Services verwenden QoS. Dabei wird QoS unterschiedlich weit gefasst, aber nur selten eindeutig definiert [BHMS05], [MN07], [Ra03].

Darum soll an dieser Stelle zwischen Qualität und QoS unterschieden werden. Der Begriff QoS entstammt dem Bereich der Netzwerke und Telekommunikation und beinhaltet bestimmte Netzwerkcharakteristika, wie z. B. Latenz und Fehlerrate. Im Bereich der SOA werden unter QoS sowohl die Eigenschaften des Netzwerkes, als auch die des Service subsummiert. Ein Service (Dienst) ist die gekapselte Funktionalität die von einer Komponente anderen Komponenten zur Verfügung gestellt wird [HMRZ06], [KT02]. Eine mögliche Definition von QoS, die sich aus der allgemeinen Qualitätsdefinition herleitet, ist demnach, QoS als der Grad, zu dem die Merkmale eines Service die Anforderungen seitens des Nutzers erfüllen. QoS wird in der Regel auf nichtfunktionale Eigenschaften begrenzt [MN07], [Ru06], [ZBNDK04], [ZN06].



Zu diesen gehören u.a. die Performance, welche wiederum Kriterien wie Latenz und Durchsatz beinhaltet, die Verfügbarkeit und die Sicherheit des Service [JMG05], [MN07], [Ru06], [ZBDKS03], [ZN06]. Zu beachten ist in diesem Zusammenhang, dass z. B. im Rahmen der Performance eine Aggregation der Eigenschaften des Netzwerks und des Web Service stattfindet. So setzt sich die Latenz aus der Zeit zusammen, die die Daten einerseits benötigen um bidirektional transportiert zu werden und zum anderen aus der Bearbeitungszeit innerhalb des Web Service. QoS-Kriterien können in Untermengen zusammengefasst werden. Man unterscheidet in laufzeit-bezogene, transaktionsunterstützungs-bezogene, konfigurationsmanagement- und kosten-bezogene, sowie sicherheits-bezogene QoS [Ra03]. Auch hier findet sich die Problematik, der uneinheitlichen Abgrenzung. Aktuell gibt es hierzu noch kein einheitliches, allgemein akzeptiertes Modell für Web Services. Eine umfangreiche Übersicht zu QoS-bezogenen Web Service Architekturen, QoS-Ontologien, QoS-Spezifikationssprachen und Klassifikationen findet sich in [ZN06]. Eine Standardisierung ist insofern wünschenswert, als damit eine vergleichende Bewertung von Web Services gefördert würde.

Wird die Qualität bei Web Services der QoS gegenübergestellt, so zeigt sich, dass diese Begriffe keinesfalls synonym zu verwenden sind. Zusammenfassend ist zu sagen, dass die QoS lediglich die nichtfunktionalen Aspekte betrachtet, während eine umfassende Qualitätsbetrachtung sowohl QoS, als auch die Qualität der funktionalen Merkmale mit einbezieht. Dieser Zusammenhang wird in Abbildung 2 verdeutlicht.

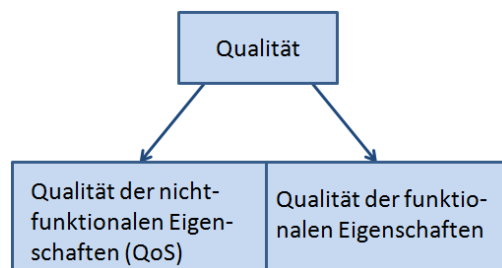


Abbildung 2: Zusammenhang zwischen Qualität und QoS

## 2.5 Agenten

Für die Realisierung der automatisierten Suche und Auswahl von geeigneten Web Services bietet sich die Agententechnologie an. Als Agent wird eine Software bezeichnet, die selbstständig und unabhängig handelt. Sie nutzt Methoden der künstlichen Intelligenz um auf flexible und autonome Weise ihre Ziele zu erreichen [CZCCL04]. Dabei befindet sich der Agent in der Interaktion mit seiner Umwelt, die durch andere Agenten, durch Nutzer oder auch durch das Internet gebildet werden kann. Agenten sind in der Lage Objekte zu manipulieren [HMRZ06].

Grundsätzlich ist die Agententechnologie in der Lage sowohl auf der Seite des Web Service Anbieters, als auch auf der Seite des Nutzers deren Aufgaben zu übernehmen. Es können sämtliche Schritte des Kaufprozesses, wie etwa Suche geeigneter Angebote, deren Bewertung und Kaufabschluss, als auch Angebotserstellung, Verbreitung desselben und Auftragsbearbeitung, mit ihrer Unterstützung umgesetzt werden. De facto treten dabei zwei Probleme auf. Zum einen bei der Prüfung der Vertrauenswürdigkeit von Angebot und Auftrag. Hierfür ist eine mögliche Lösung der Einsatz von Reputationsmechanismen. Zum anderen sind Agenten aktuell keine Rechtspersonlichkeiten nach Deutschem Recht und dürfen demnach auch keine rechtsgültigen Geschäfte tätigen [HMRZ06].

Nichts desto trotz, sind sie für die Suche nach passenden Web Services geeignet. Sie können ihre Ergebnisse dem Nutzer präsentieren, der dann letztendlich die Entscheidung über den Kauf selbst trifft [CZCCL04].

## 2.6 Multi-Attributive Nutzentheorie

Die MAUT (Multi-Attributive Utility Theory) findet Anwendung als Lösungsverfahren bei multikriteriellen Entscheidungsproblemen unter Sicherheit. Es ermöglicht eine Bewertung der Auswahlmöglichkeiten unter Zuhilfenahme von aggregierten Nutzenwerten je Auswahlmöglichkeit und Kriterium. Die Gewichte der Kriterien werden als Austauschraten zwischen diesen interpretiert.

Um den Gesamtnutzen  $n_i$  einer Alternative A zu ermitteln, werden erst die Teilnutzen  $n_j(a_{ij})$  mit den Gewichten  $w_j$  multipliziert und dann addiert [RE02].

$$n_i = \sum_{j=1}^J n_j(a_{ij}) \cdot w_j$$

Sowohl Teilnutzenfunktionen und auch Gewichte sind vor der Anwendung durch geeignete Verfahren zu ermitteln. Eine Bedingung für die Anwendung der MAUT ist die Präferenzunabhängigkeit. Durch sie ist es gestattet eine additive Gesamtnutzenfunktion anzunehmen. Ein beliebtes Beispiel für Präferenzabhängigkeit ist die Entscheidung bei einem Restaurantbesuch zwischen Rotwein und Weißwein, sowie Steak und Fisch. Hier sind die Präferenzen bezüglich der Getränke abhängig von den Hauptgerichten. Allerdings gibt es auch Autoren, die davon ausgehen, dass allein die Transformation der Ausprägungen in dimensionslose Nutzenwerte die additive Gesamtnutzenfunktion rechtfertigt [Ro98].

## 3 Vorgehensmodell zur Auswahl von Web Services

Um eine strukturierte Herangehensweise an das Auswahlproblem zu ermöglichen, wurde ein allgemeines Vorgehensmodell entworfen. Dieses ist in Abbildung 3 grafisch dargestellt. Es besteht aus fünf Phasen, die Folgendes beinhalten:

1. Spezifikationsphase: Die Spezifikationsphase beinhaltet die Festlegung sowohl der funktionalen, wie auch der nicht-funktionalen Kriterien. Diese bilden die Grundlage für die Auswahl von Web Services.
2. Ermittlungsphase: Um dem Nutzer einen geeigneten Web Service offerieren zu können, ist es notwendig geeignete Bewertungsmaßstäbe zu haben. Diese bilden in irgendeiner Form die Präferenzen des Nutzers wieder.
3. Suchphase: In dieser Phase werden die Web Services gesucht, welche die Funktionalität bieten, die der Nutzer wünscht. Dies kann automatisiert realisiert werden, in dem ein Agent die Suche in verschiedenen Discovery Services vornimmt.
4. Rankingphase: Die gefundenen Web Services, werden mit Hilfe der spezifizierten Kriterien und der ermittelten Bewertungsmaßstäbe bewertet. Dieser Schritt kann auch vom Agenten übernommen werden.
5. Abschlussphase: Abschließend werden die nach Nutzen geordneten Web Services dem Nutzer präsentiert. Aus diesen kann er wählen welchen er tatsächlich aufrufen möchte.

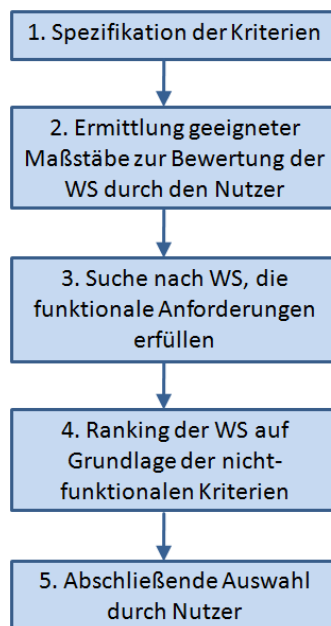


Abbildung 3: Allgemeines Vorgehensmodell zur Web Service Auswahl

### 3.1 Entscheidungsproblematik

Die Zielsetzungen, unter denen eine Entscheidung stattfindet, sind unterschiedlicher Art. Grundsätzlich wird zwischen vier Zieltypen unterschieden [RE02]:

- *Extremierungsziele*: Der Entscheider sucht ein Minimum oder Maximum der Zielausprägung.
- *Satisfizierungsziele*: Es werden Werte der Zielvariablen angestrebt, die eine bestimmte Schwelle über- oder unterschreiten.
- *Fixierungsziele*: Es sind konkrete Werte der Zielausprägungen vorgegeben, die erreicht werden sollen. Auch Intervalle sind möglich.
- *Approximierungsziele*: Bei Zielen dieser Art ist der Abstand zwischen Zielausprägung und Wertvorgabe zu minimieren.

Anforderungen, die lediglich erfüllt oder nicht erfüllt werden können, werden auch als Fixierungsziele betrachtet, mit der binären Ausprägung  $[0; 1]$ . Das heißt bietet z. B. ein Web Service einen nachgefragten Verschlüsselungsalgorithmus an, wird dieses Ziel mit 1 bewertet, ansonsten mit 0.

Herden et al. führt ein auf entscheidungstheoretischen Grundlagen basierendes Lebenslagenmodell ein. Angenommen wird, dass sich ein Kunde in einer Entscheidungssituation befindet, wenn er Informationen nachfragt. Es sei eine Entscheidungssituation  $s \in S$ , in der die Menge aller gültigen Handlungsalternativen als  $A_z \subseteq A$  definiert sei. Zudem existiere für alle  $a_z \in A_z$  eine Menge von Eigenschaften  $e_{az} \subseteq E$ , auch als Attribute der Handlungsalternativen bezeichnet. Jedes Tripel  $l = (s, a_z, e_{az})$  definiert genau eine Lebenssituation eines Entscheidungsträgers. Eine Lebenslage wiederum besteht aus einer endlichen Menge von Lebenssituationen.

Ein Web Service kann als vollständig durch seine Eigenschaften  $e_1 \dots e_n$  beschrieben betrachtet werden [LASG07]. Befindet sich der Nutzer nun in einer Situation  $s$ , in der es konkret um die Auswahl eines Web Service aus einer endlichen Menge von Web Services geht, so kann die Menge aller Web Services, welche die geforderte Funktionalität besitzen, als  $a_z$  und somit die Menge all ihrer Eigenschaften als  $e_{az}$  betrachtet werden.  $e_{az}$  wird auch bezeichnet als die Konfiguration des Web Service. Somit entspricht die Auswahl von Web Services aus Sicht der Entscheidungstheorie einem multikriteriellen Entscheidungsproblem bei Sicherheit, wenn unterstellt wird, dass dem Entscheider alle für ihn relevanten Informationen bezüglich der entsprechenden Optionen bekannt sind. Dies kann durch die umfassende Spezifikation von Web Services realisiert werden. Bei der Auswahl von Web Services geht es dem Nutzer nicht notwendigerweise immer um den objektiv besten Web Service. Vielmehr ist für ihn ein Ergebnis von Relevanz, das ihm den größtmöglichen Nutzen liefert. Das globale Ziel bei der Entscheidungsunterstützung muss also sein, einen Web Service zu finden, der den Nutzen des Nutzers maximiert.

Ein Verfahren zur Lösung des Problems ist die Multi-Attributive Nutzentheorie. Sie wurde gewählt, weil sie zum einen die Teilnutzen der Kriterien berücksichtigt und zum anderen die Gewichte in direktem Zusammenhang mit der Nutzenbewertung stehen. Dadurch bekommen sie die Bedeutung von Austauschraten zwischen den Kriterien. Besondere Aufmerksamkeit bei Anwendung der MAUT verdienen die Fixierungsziele. Da die MAUT originär mit einer additiven Gesamtnutzenfunktion arbeitet, müssen die Nutzen der Fixierungsziele mit binärer Ausprägung als Multiplikatoren in die Nutzenberechnung aufgenommen werden. Dabei werden diese Nutzen ohne Gewichte verwendet. Der Nutzen  $n_f$  eines Fixierungszieles  $f$  ist gleich dem Binärwert des Zieles. Es ergibt sich folgende geänderte Gesamtnutzenfunktion:

$$n_{az} = \sum_{i=1}^n n_i(e_{azi}) \cdot w_i \cdot \prod_{f=1}^m n_f$$

Dadurch ist sichergestellt, dass der Gesamtnutzen einer Alternative null ist, wenn auch nur eines der Fixierungsziele nicht erfüllt wird.

Wenn die MAUT angewendet wird, müssen in der Ermittlungsphase des Modells die partiellen Nutzenfunktionen und die Kriteriengewichte mit geeigneten Verfahren bestimmt werden. In der vierten Phase kann dann unter Anwendung der MAUT eine Rangordnung der gefundenen Web Services erstellt werden. Hierzu wird der Gesamtnutzen einer Alternative durch Anwendung der geänderten Nutzenfunktion ermittelt. Die so geordneten Web Services können dem Nutzer zur endgültigen Entscheidung vorgelegt werden.

**Sonderfall 1:** Ist der Nutzer in der Entscheidungssituation nun in der Lage, für alle der für ihn relevanten Eigenschaften  $e_1 \dots e_n$  konkrete Ausprägungen vorzugeben, so liegen nur noch Satisfizierungs- und Fixierungsziele vor. Damit hat der Nutzer die Konfiguration  $e_u$  definiert, die er bereit ist zu akzeptieren. Vor dem betriebswirtschaftlichen Hintergrund der Kostenminimierung betrachtet, heißt das, dass der Preis als zu minimierendes Extremierungsziel betrachtet wird. Wird nun zusätzlich unterstellt, dass eine Übererfüllung der Satisfizierungsziele keinen zusätzlichen Nutzen bringt, so wird das multikriterielle Entscheidungsproblem reduziert auf ein Entscheidungsproblem mit einem Kriterium. Vorausgesetzt wird, es existiert eine nicht leere Menge von Web Services  $A_z \subseteq A$ , deren Elemente  $a_z$  mit ihren dazugehörigen Konfigurationen  $e_{az}$ , die durch  $e_u$  formulierten Bedingungen zumindest erfüllen. Zudem wird unterstellt eine Übererfüllung einzelner Konfigurationselemente bringt keinen zusätzlichen Nutzen für den Nutzer. Es wird der günstigste Web Service  $a_b = \min \{ p(a_z) \mid a_z \in A \}$  gewählt.

Entsteht hingegen durch eine Übererfüllung einzelner Kriterien ein zusätzlicher Nutzen, so kann das Problem nicht derart vereinfacht werden in diesem Fall wiederum die MAUT angewendet.

**Sonderfall 2:** Hier wird angenommen, der Nutzer spezifiziert überhaupt keine Präferenzen. Ein möglicher Lösungsansatz für dieses Problem lässt sich mit Hilfe von Agenten realisieren. Diese ordnen zunächst den gewünschten oder benötigten Web Service anhand der nachgefragten Funktionalitäten in einen spezifischen Kontext ein. Aufgrund des Kontextes wird nun den Anforderungen ein vordefiniertes QoS-Profil beigelegt, welches vorgegebene QoS-Kriterien mit konkreten Ausprägungen enthält. Anschließend wird unter Zuhilfenahme der Kriterien aus dem QoS-Profil die Menge der Web Services bestimmt, welche die Anforderungen erfüllt. So ist im Kontext von Finanztransaktionen ein QoS-Profil denkbar, das besonderen Wert auf Sicherheit legt. Das können zum Beispiel eine starke Verschlüsselung und bestimmte Authentifizierungsmechanismen sein. Diese QoS-Profile könnten beispielsweise vom Service Provider oder einer unabhängigen Institution als Empfehlungen im Discovery Service als QoS-Standard hinterlegt werden.

Anhand der vorgegebenen QoS-Kriterien wird aus den Web Services mit entsprechender Funktionalität der Günstigste ausgewählt.

#### **4 Fazit**

Auch mehrere Jahre nach dem Erscheinen der ersten Publikationen zum Thema Web Services gibt es in wichtigen Punkten dieses Themas noch keine einheitlichen Standards. Es existieren mittlerweile genug Möglichkeiten um Web Services ausreichend zu spezifizieren. Aber keiner dieser Vorschläge hat sich bisher etabliert. Auch ist der Begriff QoS nicht eindeutig umrissen und wird unterschiedlich weit gefasst. Den Anspruch diese Problematik zu lösen hat das vorliegende Paper nicht.

Das hier eingeführte Vorgehensmodell bietet dem Nutzer einen flexiblen Rahmen zur Lösung des Auswahlproblems. Es ist erweiterbar und kann mit verschiedenen Technologien umgesetzt werden. Durch den Einsatz von Software-Agenten kann der Ablauf zu großen Teilen automatisiert werden.

Wenn auch vorgeschlagen wird, mit der Multi-Attributiven Nutzentheorie die Auswahlmöglichkeiten in eine Präferenzordnung zu bringen, so darf trotz dessen nicht der wirtschaftliche Aspekt vernachlässigt werden. Die Ermittlung der Nutzenfunktionen und der Gewichte, aber auch die Prüfung der Präferenzunabhängigkeit sind aufwändige Verfahren. Unter Einbeziehung dieser Tatsachen ist es notwendig einen Trade-off zwischen dem zur Nutzenberechnung benötigten Aufwand und dem sich daraus ergebenden Vorteil zu suchen.

## Literaturverzeichnis

- [AM07a] Al-Masri, E.; Mahmoud, Q. H.: Crawling multiple UDDI business registries. In: Proceedings of the 16th international conference on World Wide Web. ACM, New York 2007, 1255-1256.
- [AM07b] Al-Masri, E.; Mahmoud, Q. H.: Discovering the Best Web Service. In: Proceedings of the 16th international conference on World Wide Web. ACM, New York 2007, 1257-1258.
- [ACKM04] Alonso, G.; Casati, F.; Kuno, H.; Machiraju, V.: Web Services – Concepts, Architectures and Applications. Springer, Berlin 2004.
- [Ba98] Balzert, H.: Lehrbuch der Software-Technik : Software-Management. Software-Qualitätssicherung, Unternehmensmodellierung. Spektrum, Heidelberg 1998.
- [BHMS05] Berbner, R.; Heckmann, O.; Mauthe, A.; Steinmetz, R.: Eine Dienstgüte unterstützende Webservice-Architektur für flexible Geschäftsprozesse. In: Wirtschaftsinformatik 47 nr. 4 (2005), 268-277.
- [CZCCL04] Cao, J.; Zhou, J.; Chen, D.; Chan, A. T. S.; Lu, J.: Mobile Agent Technology for Web Applications. In: Zhang, Y.-Q.; Kandel, A.; Lin, T. Y.; Yao, Y. Y. (Hrsg.): Computational Web Intelligence: Intelligent Technology for Web Applications. World Scientific, Singapur 2004. 453-479.
- [CEN05] CEN (Hrsg.): DIN EN ISO 9000:2005 Qualitätsmanagementsysteme – Grundlagen und Begriffe. Brüssel 2005.
- [Ce02] Cerami, E.: Web Service Essentials. O'Reilly, Sebastopol 2002.
- [Do07] Dobson, Glen: Quality of Service in Service-Oriented Architectures. <http://digs.sourceforge.net/papers/qos.html> 18. September 2007.
- [DGH03] Dustdar, S.; Gall, H.; Hauswirth, M.: Software-Architekturen für Verteilte Systeme. Springer, Berlin 2003.
- [Ga84] Garvin, D. A.: What Does "Product Quality" Really Mean? In: Sloan Management Review 1984, Vol. 26 Issue 1, 25-43.
- [HMRZ06] Herden, S.; Marx Gómez, J.; Rautenstrauch, C.; Zwanziger, A.: Software-Architekturen für das E-Business. Springer, Berlin 2006.
- [JMG05] Jaeger, M.C.; Mühl, G.; Golze, S.: QoS-Aware Composition of Web Services: An Evaluation of Selection Algorithms. In: On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE. Springer, Berlin 2005, 646-661.
- [Ju98] Juran, J. M.: How to think about Quality. In: Juran, J.M.; Godfrey, A.B. (Hrsg.): Juran's quality handbook. McGraw-Hill, New York 1998. 2.1-2.18.
- [KT02] Krammer, A.; Turowski, K.: Spezifikation von Web Services. In: Erich Ortner; Sven Overhage (Hrsg.): 1. Workshop „Entwicklung von Anwendungen auf Basis der XML Web-Service-Technologie“. Darmstadt 2002, 1-16.
- [KST02] Kulchenko, P.; Snell, J.; Tidwell, D.: Programming Web Services with SOAP. O'Reilly, Sebastopol 2002.

- [LASG07] Lamparter, S.; Ankolekar, A.; Studer, R.; Grimm, S.: Preference-based Selection of Highly Configurable Web Services. In: Proceedings of the 16th international conference on World Wide Web. ACM, New York 2007, 1013-1022.
- [MN07] Mani, A.; Nagarajan, A.: Understanding quality of service for Web services. <http://www.ibm.com/developerworks/library/ws-quality.htm>. 15. Juli 2007.
- [OT05] Overhage, S.; Thomas, P.: WS-Specification: Ein Spezifikationsrahmen zur Beschreibung von Web Services auf Basis des UDDI-Standards. In: Ferstl, O. K. et al. (Hrsg.): Wirtschaftsinformatik 2005. Heidelberg 2005, 1539-1558.
- [Ra03] Ran, S.: A Model for Web Services Discovery With QoS. In: ACM SIGecom Exchange, Vol. 4, Issue 1 (2003), 1-10.
- [RE02] Rommelfanger, H. J.; Eickemeier, S. H.: Entscheidungstheorie : Klassische Konzepte und Fuzzy-Erweiterungen. Springer, Berlin 2002.
- [Ro98] Roth, B.: Lösungsverfahren für mutlikriterielle Entscheidungsprobleme : klassische Verfahren, neuronale Netze und Fuzzy logic. Lang, Frankfurt a. M. 1998.
- [Ru06] Rud, D.: Qualität von Web Services. Messung und Sicherung der Performance. Vdm Verlag, Saarbrücken 2006.
- [TPPE05] Tomic, V.; Pagurek, B.; Patel, K.; Esfandiari, B.; Ma, W.: Management applications of the Web Service Offerings Language (WSOL). In: Information Systems 30, Issue 7 (2005), 564-586.
- [W3C07a] W3C: Web Service Architecture. <http://www.w3.org/TR/ws-arch/>. 22. August 2007.
- [W3C07b] W3C: Semantic Annotations for WSDL Working Group. <http://www.w3.org/2002/ws/sawsdl/>. 22. August 2007.
- [VHA05] Vu, L.-H.; Hauswirth, M.; Aberer, K.: QoS-based Service Selection and Ranking with Trust and Reputation Management. In: Proceedings of the OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2005 Proceedings, Part I, vol. 3760, num. 1 2005, 446-483.
- [ZBDKS03] Zeng, L.; Benatallah, B.; Dumas, M.; Kalagnanam, J.; Sheng, Q. Z.: Quality Driven Web Services Composition. In: Proceedings of the 12th international conference on World Wide Web. ACM, New York 2003. 411-421.
- [ZBNDK04] Zeng, L.; Benatallah, B.; Ngu, A. H. H.; Dumas, M.; Kalagnanam, J.; Chang, H.: QoS-Aware Middleware for Web Services Composition. In: IEEE Transactions on Software Engineering, Vol. 30, No. 5 (2004), 311-327.
- [ZN06] Zhou, J.; Niemelä, E.: Toward Semantic QoS Aware Web Services: Issues, Related Studies and Experience. In: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence. IEEE Computer Society 2006. 553-557.