

# **Diplomarbeit**

## Ein Referenzmodell für Profiling bei e-Business- Websites

Matthias Dorka

**Universität Dortmund**

Fachbereich Informatik

2001-07-16

Gutachter:

Prof. Dr. Volker Gruhn

Lehrstuhl X - Software-Technologie

Prof. Dr. Heiko Krumm

Lehrstuhl IV - Rechnernetze und verteilte Systeme

## **ZUSAMMENFASSUNG**

In der vorliegenden Diplomarbeit wird ein Referenzmodell entworfen, welches die Gewinnung, Haltung und Auswertung von individuellen Konsumentenprofilen ermöglicht, wie sie bei der elektronisch unterstützten Geschäftsabwicklung im Internet verwendet werden können.

Dazu wird in Kapitel 1 das zugrunde liegende Thema motiviert und eine Einordnung in vorhandene Literatur sowie verwandte Forschungs- und Anwendungsgebiete vorgenommen. Anschließend erfolgt in Kapitel 2 die umfassende Definition von Anforderungen an Profiling, welche neben Gewinnung, Haltung und Auswertung auch Aspekte der Visualisierung von Profildaten betreffen. Dabei wird ebenfalls die Notwendigkeit einer Erfolgskontrolle und der Anpassung des Profilings an das Geschäftsmodell des e-Business-Anbieters diskutiert.

Kapitel 3 stellt vorhandene Internet- und Server-Technologien sowie Standards zur Realisierung von Profilingmaßnahmen vor, die gemeinsam mit den Anforderungen aus Kapitel 2 in die Erstellung eines universellen Objektmodells in Kapitel 4 einfließen.

Eine Validierung dieses Modells findet in Kapitel 5 durch dessen beispielhafte Instanziierung bei zwei existierenden e-Business-Websites statt: der Community-Plattform c-lounge und dem Online-Versicherungsmakler Quinner.

Abschließend erfolgt in Kapitel 6 eine zusammenfassende Bewertung der Ergebnisse unter Hinweis auf weiterführende Aufgaben. Dabei wird auch ein Ausblick auf eine mögliche Implementierung des Entwurfs gegeben.

## INHALTSVERZEICHNIS

<b>1</b>	<b>EINLEITUNG</b>	<b>8</b>
1.1	<b>Einführung und Motivation</b>	<b>8</b>
1.1.1	Electronic Business und Electronic Commerce	8
1.1.2	Communities	10
1.1.3	Portale	11
1.1.4	Personalisierung und Profiling	11
1.1.5	Ziel der Diplomarbeit	13
1.2	<b>Kontext der Diplomarbeit</b>	<b>14</b>
1.2.1	Literatur	14
1.2.2	Produkte	16
1.2.3	Anwendung	17
1.2.4	Verwandte Methoden und Techniken	18
<b>2</b>	<b>DEFINITION VON ANFORDERUNGEN AN PROFILING</b>	<b>22</b>
2.1	<b>Anforderungen an die Profildatengewinnung</b>	<b>22</b>
2.1.1	Nutzererfassung	22
2.1.2	Einzel- und Gruppenprofile	23
2.1.3	Profildatengewinnung	24
2.1.4	Integrationsmöglichkeit für Profilinformatoren anderer Kommunikationskanäle	26
2.1.5	Einfache und individuelle Anpassbarkeit der Profiling-Intensität	27
2.1.6	Privatsphäre und Datenschutz	27
2.2	<b>Anforderungen an die Profildatenhaltung</b>	<b>29</b>
2.2.1	Grundlagen	29
2.2.2	Erweiterbarkeit, Strukturierung und Werteskalierung	29
2.2.3	Möglichkeiten des Datenaustauschs	32
2.3	<b>Anforderungen an die Profildatenauswertung</b>	<b>34</b>
2.3.1	Profilnutzung und -auswertung zur Laufzeit	34
2.3.2	Operationen auf Profilen	34
2.3.3	Analyse von Profilen	35
2.4	<b>Anforderungen an die Profildatenvisualisierung</b>	<b>38</b>
2.4.1	Clickstreams	38
2.4.2	Einzelprofile	39
2.4.3	Profilmengen	40
2.5	<b>Übergeordnete Anforderungen</b>	<b>42</b>
2.5.1	Anpassungsfähigkeit an Geschäftsmodelle	42
2.5.2	Erfolgskontrolle des Profiling	44
2.5.3	Technische Integrationsmöglichkeit	45

2.5.4	Performanz und Skalierbarkeit	45
<b>3</b>	<b>TECHNISCHE GRUNDLAGEN FÜR PROFILING</b>	<b>46</b>
<b>3.1</b>	<b>Internettechniken als Grundlage der Kommunikation</b>	<b>46</b>
3.1.1	Das Hypertext Transfer Protocol	46
3.1.2	Cookies	48
3.1.3	Session-IDs	49
3.1.4	Web Bugs	49
<b>3.2</b>	<b>Servertechniken für die Profildatenverarbeitung und –speicherung</b>	<b>51</b>
3.2.1	Serverseitige Dynamik	51
3.2.2	Realzeit- und Vorabauswertungen	53
3.2.3	Lastverteilung	54
<b>3.3</b>	<b>Standards im Kontext von Profiling</b>	<b>56</b>
3.3.1	Platform for Privacy Preferences	56
3.3.2	Electronic Commerce Modeling Language	58
3.3.3	Customer Profile Exchange	58
3.3.4	Open Profiling Standard	60
<b>3.4</b>	<b>Metainformationen für Elemente einer Website</b>	<b>61</b>
3.4.1	Grundlagen	61
3.4.2	Zuordnung von Metainformationen zu statischem HTML	67
3.4.3	Zuordnung von Metainformationen zu dynamisch erzeugtem HTML	68
3.4.4	Zuordnung von Metainformationen zu Transaktionen	69
<b>4</b>	<b>EIN MODELL FÜR PROFILING</b>	<b>71</b>
<b>4.1</b>	<b>Objektorientierte Analyse</b>	<b>71</b>
4.1.1	Anwendungsfälle	71
4.1.2	Grundlegende Objekte	73
<b>4.2</b>	<b>Entwurf eines Modells für Profiling</b>	<b>78</b>
4.2.1	Unterstützende Objekte und Objektverteilung	78
4.2.2	Interaktionen zwischen Objekten	87
4.2.3	Zusätzliche Komponenten	93
4.2.4	Zusammenfassung und Überblick	103
<b>5</b>	<b>INSTANZIIERUNG DES MODELLS AUF ANWENDUNGSBEISPIELE</b>	<b>105</b>
<b>5.1</b>	<b>Die Community-Plattform c-lounge</b>	<b>105</b>
5.1.1	Geschäftsmodell von c-lounge	105
5.1.2	Profiling bei c-lounge	106
5.1.3	Beschreibung der Webanwendung von c-lounge	107
5.1.4	Einsatz des Referenzmodells für Profiling bei c-lounge	113

---

5.1.5	Bewertung und Ausblick	121
<b>5.2</b>	<b>Quinner</b>	<b>124</b>
5.2.1	Geschäftsmodell von Quinner	124
5.2.2	Profiling bei Quinner	125
5.2.3	Beschreibung der Webanwendung von Quinner	126
5.2.4	Einsatz des Referenzmodells für Profiling bei Quinner	128
5.2.5	Bewertung und Ausblick	129
<b>6</b>	<b>FAZIT UND AUSBLICK</b>	<b>131</b>
6.1	Anforderungen und Entwurf	131
6.2	Einsetzbarkeit und Nutzen	132
6.3	Vision einer Realisierung	133
	<b>ABBILDUNGSVERZEICHNIS</b>	<b>135</b>
	<b>TABELLENVERZEICHNIS</b>	<b>136</b>
	<b>ABKÜRZUNGSVERZEICHNIS</b>	<b>137</b>
	<b>GLOSSAR</b>	<b>139</b>
	<b>LITERATURVERZEICHNIS</b>	<b>143</b>

## **1 Einleitung**

In diesem Kapitel erfolgt zunächst eine Einführung in das Themengebiet Electronic Business und eine Motivation der Zielsetzung dieser Arbeit. Anschließend wird Profiling in einen Kontext hinsichtlich vorhandener Literatur, existierender Softwareprodukte, bisheriger Anwendung und angrenzender technischer Bereiche eingebettet.

### **1.1 Einführung und Motivation**

Zur Einführung in die Thematik des Profilings, welche das Gewinnen, Halten und Auswerten von Konsumentenprofilen bezeichnet, werden grundlegende Begriffe wie Electronic Business und Electronic Commerce erläutert. Darüber hinaus werden Communities und Portale im Hinblick auf die späteren Beispielanwendungen des zu erstellenden Referenzmodells im Besonderen betrachtet; diese Ausprägungen von Websites sind jedoch auch zum Verständnis aktueller Entwicklungstendenzen elektronischer Märkte von Bedeutung.

#### **1.1.1 Electronic Business und Electronic Commerce**

Die Bezeichnungen Electronic Business (e-Business, EB) und Electronic Commerce (e-Commerce, EC) unterliegen zurzeit noch keiner allgemein verbreiteten Definition und werden in vielfältiger Weise und z. T. synonym verwendet [HeSa99].

In letzter Zeit häuft sich in Wissenschaft und Wirtschaft der Einsatz der Begriffe dahingehend, dass unter e-Business alle verteilten Geschäftsprozesse einer Organisation verstanden werden, deren Ablauf auf dem Einsatz elektronischer Medien basiert, während e-Commerce den Kauf und Verkauf von Waren und Dienstleistungen über elektronische Medien subsumiert.

Eine neuere Definition aus [KRT99] stellt e-Commerce als die Nutzung elektronischer Medien für traditionelle Geschäftsmodelle dar, während e-Business die Anwendung neuartiger Geschäftsmodelle bezeichnet, welche erst durch die veränderten Regeln einer informationstechnisch vernetzten Wirtschaft ermöglicht werden.

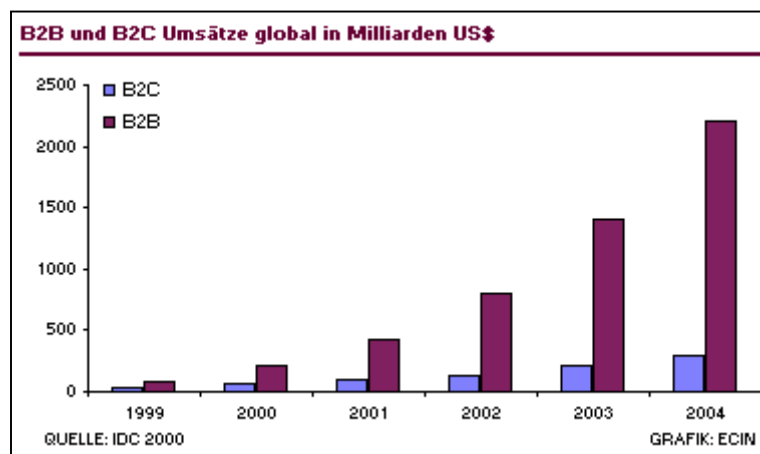
Innerhalb des Bereichs e-Commerce wird zudem nach der Art der Marktteilnehmer elektronisch unterstützter Transaktionen unterschieden, wie dies in Tabelle 1 dargestellt ist.

		Nachfrager der Leistung		
		Consumer	Business	Administration
Anbieter der Leistung	Consumer	Consumer-to-Consumer (C2C)	Consumer-to-Business (C2B)	Consumer-to-Administration (C2A)
	Business	Business-to-Consumer (B2C)	Business-to-Business (B2B)	Business-to-Administration (B2A)
	Administration	Administration-to-Consumer (A2C)	Administration-to-Business (A2B)	Administration-to-Administration (A2A)

**Tabelle 1: Markt- und Transaktionsbereiche des e-Commerce [HaSe99]**

Von besonderer Bedeutung sind dabei die Bereiche Business-to-Consumer (B2C) und Business-to-Business (B2B). "Obwohl bisher die spektakulärsten Electronic Commerce-Erfolgsbeispiele des Business-to-Consumer-Bereiches (...) von sich Reden machen, ist es heute und auch in Zukunft der Business-to-Business-Bereich, der das enorme Potential des Electronic Commerce ausmacht" [HeSa99].

Belegt wird diese Aussage durch zahlreiche Studien und Prognosen von Beratungsunternehmen und Marktforschungsinstituten; so schätzt IDC die globale Umsatzentwicklung in den Bereichen B2C und B2B wie in Abbildung 1 gezeigt ein.



**Abbildung 1: Prognose der globalen Umsatzentwicklung B2C und B2B [ECIN01]**

Allerdings haben sich die optimistischen Umsatzprognosen der letzten Jahre, insbesondere für den B2C-Bereich in Europa, bis heute kaum bestätigt. Laut einer Forrester-Studie liegen die Gründe hierfür "in erster Linie in der nur langsam voranschreitenden Deregulierung und Liberalisierung der europäischen Telekommunikationsmärkte, der mangelnden Technologiefreundlichkeit, den unbefriedigenden infrastrukturellen Gegebenheiten und dem nationalen bzw. europäischen regulatorischen Umfeld" [HeSa99].

Aber auch die Ertragssituation der erfolgreich in diesem noch recht kleinen Markt positionierten Unternehmen ist in den meisten Fällen als schlecht einzustufen, da die hohe Zahl der Konkurrenten und die durch das Medium Internet bedingte Preistransparenz zu hoher Wettbewerbsintensität führen. Als viel versprechende Wege aus dieser Krise gelten der Aufbau einer (Online-)Marke, die Schaffung von Mehrwertnutzen für den Kunden sowie die Personalisierung von Inhalten und Produkten, um eine Differenzierung gegenüber den Wettbewerbern zu erreichen, die im Idealfall mit der Implementierung eines elektronisch unterstützten *Customer Relationship Managements*<sup>1</sup> (CRM) und der Verfolgung einer kundenorientierten Unternehmenspolitik bis hin zum One-to-One-Marketing einhergehen [Stol99], [Stol00].

### 1.1.2 Communities

Als beliebte Anziehungspunkte für Internetnutzer haben sich sogenannte virtuelle Gemeinschaften, im Englischen Virtual Communities oder auch kurz Communities, herausgestellt. Als solche werden "Gruppen von Nutzern mit gleichgelagerten Interessen und Bedürfnissen bezeichnet, die sich mit dem Ziel des Gedankenaustauschs und der Gemeinschaftsbildung online treffen" [TSB00].

Obwohl Communities im Ursprung nichtkommerziellen Vereinigungen Gleichgesinnter entstammen, die mithilfe von Newsgroups oder Bulletin Board Systemen hinsichtlich eines gemeinsamen Interessensgebiets kommunizierten, etablieren sich heute zahlreiche Communities um bestimmte Produktgruppen oder Anbieter und nutzen die von diesen bereitgestellte Infrastruktur für ihr virtuelles Gemeinschaftsleben. Erfolgreiches Beispiel für eine derartige Symbiose ist der Internet-Buchhändler Amazon, der seinen Kunden ermöglicht, Online-Rezensionen zu Büchern zu verfassen, was sowohl dem Mitteilungsbedürfnis der Leser entgegenkommt als auch den Rat suchenden Interessenten einen Mehrwert verschafft.

Wenngleich Amazon lediglich das Einstellen eines kurzen Texts, die freiwillige Angabe des Namens und der e-Mail-Adresse des Verfassers sowie eine Bewertung des Buches vorsieht, werden hierbei bereits nach [HaAr97] wesentliche Kennzeichen einer Community deutlich:

- Die Community organisiert sich um einen spezifischen Interessensschwerpunkt.
- Es besteht die Möglichkeit, Inhalte und Kommunikation zu integrieren.
- Dabei werden Informationen verwendet, die von den Mitgliedern bereitgestellt werden.
- Es liegt eine kommerzielle Orientierung vor.

Weiterhin zeigt sich, dass die von den Teilnehmern eingebrachten Inhalte den Nutzen und damit die Attraktivität des Online-Angebots steigern, was sowohl dem Betreiber als auch den Besuchern der Website zugute kommt.

Communities, die mehr Interaktionsmöglichkeiten bieten, z. B. virtuelle Abstimmungen (*Votings*) oder Online-Chats, und sich einem vielschichtigeren Thema widmen, beispielsweise Finanzen und Geldanlage, einer Sportart oder einem Lebensabschnitt,

---

<sup>1</sup> Nicht explizit eingeführte Begriffe, die unbekannt sein könnten, werden bei ihrem ersten Vorkommen *kursiv* dargestellt und in einem Glossar im Anhang erläutert, wo sich auch ein Abkürzungsverzeichnis befindet.



entwickeln in der Regel ein fruchtbares Eigenleben mit sozialem Charakter, sobald eine als 'kritische Masse' bezeichnete Anzahl aktiver Mitglieder erreicht ist.

Die vom Betreiber anfänglich notwendige Mitarbeit am Aufbau der Community kann reduziert oder auf Überwachungs- und Pflegearbeiten beschränkt werden, wobei auch hier die Einbeziehung der Community-Mitglieder möglich ist [Duch99].

Damit bieten Communities einen erheblichen Mehrwert für Internetnutzer gegenüber herkömmlichen Informationsangeboten und können, wenn sie vom Betreiber geschickt eingesetzt werden, ein Bindeglied zwischen Anbietern kommerzieller Angebote und potenziellen Konsumenten bilden.

Denkt man beispielsweise an Communities für börseninteressierte Privatanleger, ambitionierte Motorradfahrer oder einkommensstarke Senioren, so liegt der besondere Wert einer Community für den Betreiber in der Möglichkeit, bestimmte Waren und Dienstleistungen zu verkaufen oder gegen Provision zu vermitteln, die Fülle der von den Mitgliedern eingebrachten Informationen und Meinungen zu analysieren und im Sinne eines adaptiven Marketings in Produktverbesserungen oder -innovationen umzusetzen. Eine große Gruppe ggf. einschließlich Kontaktinformationen bekannter Personen, die ein gemeinsames Interesse teilen, bietet letztendlich auch ein gewaltiges Potenzial für fokussierte Werbung oder Marktforschung.

### 1.1.3 Portale

Eine weitere Form unter den hochfrequentierten Websites sind Portale, "die im Mittelpunkt des Interesses vieler Surfer stehen ... (und) ein 'Eingangstor zum Internet' darstellen" [Stol99]. Die etablierten horizontalen Portale entstammen oftmals Service-Websites wie Suchmaschinen oder Verzeichnisdiensten, aggregieren jedoch zunehmend weitere Funktionen wie Nachrichten- oder Messaging-Dienste und wollen mit ihrem breiten Themenspektrum erste Anlaufstelle für die Nutzung des Internets sein.

Daneben existieren zahlreiche vertikale Portale, die sich mit einem bestimmten Themenschwerpunkt befassen (z. B. Sport, Gesundheit, Geldanlage usw.) um dafür eine dienstleistungsbringende und verkehrsteuernde Funktion zu übernehmen.

Die hieraus erwachsenden Ertragspotenziale veranlassen zahlreiche Unternehmen, Websites zu erstellen und zu betreiben, die den Anspruch eines Portals erheben, deren Erfolg jedoch von einer breiten Akzeptanz unter den Nutzern abhängt und damit pro Themensegment nur eine kleine Anzahl langfristig erfolgreicher Anbieter zulässt.

Insbesondere Dienstleistungsunternehmen versuchen zunehmend, den bündelnden Charakter eines Portals für das Anbieten eigener Leistungen zu nutzen, indem sie diese zu variablen Service-Paketen zusammenfassen und auf ihren Unternehmens-Websites anbieten, die als eine ganze Branche abdeckendes Portal kommuniziert wird (z. B. Gebrauchtwagenbörsen, Online-Auktionen, Finanzdienstleister).

### 1.1.4 Personalisierung und Profiling

Zurzeit entwickeln viele Unternehmen aus dem Bereich des e-Business Strategien, ihre internetbasierte Geschäftsprozessabwicklung nicht nur unter Konsumenten zu etablieren, sondern auch rentabel zu gestalten.

Da aufgrund der hohen Preistransparenz und der Vielzahl konkurrierender Anbieter im Bereich e-Commerce das Erreichen von Wettbewerbsvorteilen durch Kostenführerschaft

nahezu unmöglich ist, wird verstärkt versucht, sich über Leistungsvorteile zu differenzieren. Die hierdurch gewonnenen Kunden sollen anschließend mittels Kundenbindungsstrategien längerfristig gehalten werden [Stol99].

Diese Leistungsvorteile können in direktem Zusammenhang mit den angebotenen Produkten stehen, indem beispielsweise die Bequemlichkeit der Kunden durch intelligente Suchfunktionen, virtuelle Verkaufsberater oder One-Click-Bestellungen für Produkte unterstützt wird. Weitere Nutzervorteile bilden die verlässliche Angabe der Verfügbarkeit von Artikeln, die Garantie kurzer Lieferzeiten sowie die Möglichkeit, den Lieferstatus einer Bestellung online zu verfolgen.

Dagegen bestehen produktunabhängige Mehrwertangebote vor allem aus kostenlosen Messaging-Diensten oder Community-Komponenten. Erstere sind für horizontale Portale typisch und umfassen e-Mail-Konten, SMS-Versand und Speicherplatz für eine private Homepage, während letztere in Form von Bulletin-Board-Systemen, Votings oder Chats das interpersonelle Kommunikations- und Entertainment-Bedürfnis der Internetnutzer befriedigen.

Eine gute Möglichkeit, die sogenannte Convenience einer Website, also deren bequeme Benutzbarkeit, zu steigern, bietet die Personalisierung. Darunter ist die individuelle Anpassung einer Website an einzelne Nutzer(-gruppen) zu verstehen. Diese kann durch den Nutzer selbst veranlasst werden oder automatisiert geschehen und in Anlehnung an [Spil00] den redaktionellen Inhalt, das Seitendesign und die Site-Struktur betreffen.

Portale wie MSN (<http://www.msn.de>) oder MyYahoo! (<http://my.yahoo.com>) erlauben es beispielsweise, eine persönliche Startseite durch den Nutzer dahin gehend zu gestalten, dass dieser Person nur besonders interessante Nachrichtenmeldungen bestimmter Themengebiete wie Politik, Wirtschaft, Sport, die lokale Wettervorhersage oder ausgewählte Aktienkurse angezeigt werden, wobei das Layout der Seite bezüglich Anordnung der *Content*-Bereiche und Farbgebung angepasst werden kann.<sup>2</sup>

[MAB00] definiert weitgefasst: "Personalization is the provision to the individual of tailored products, services, information or information relating to products or service" und nähert sich damit den grundlegenden Prinzipien des Direktbeziehungsmarketings und des darauf aufbauenden One-to-One-Marketings, welche eine individualisierte Kundenkommunikation, aber auch eine an den einzelnen Konsumenten angepasste Produkt-, Preis- und Distributionspolitik in den Mittelpunkt stellen, und damit wesentlich auf die im e-Business erwünschte Kundenbindung abzielen [Röde00].

Basis dieses Ansatzes ist, genau wie beim Database- und Direct Marketing als Ursprüngen dieser Denkrichtung, eine umfassende Kunden- und Interessentendatenbank, die neben üblichen Stammdaten wie Name, Anschrift, usw. auch sogenannte Potenzial- und Reaktionsdaten enthält. Beispielsweise sind dies Informationen über das bisherige Kaufverhalten oder von Kunden bekundetes Interesse an bestimmten Produkten [KoBl99], [Lin00].

---

<sup>2</sup> Besonders anschauliche Beispiele, die für das Verständnis des Textes nicht unbedingt erforderlich sind, werden nachfolgend grau hinterlegt und gerahmt dargestellt. Eigene Definitionen sind dagegen durch weiße Rahmen kenntlich gemacht, wobei der festzulegende Begriff **fett** gesetzt ist.

Damit konvergieren die oben dargelegten Ansätze der Personalisierung und des Marketings in ihrem gemeinsamen Bedarf nach Profildaten, die geeignet sind, ein an Austauschprozessen zwischen Marktteilnehmern beteiligtes Individuum in nutzbringendem Maße zu charakterisieren.

Ein **Profil** kann somit zunächst als eine Ansammlung von Daten verstanden werden, die einem Individuum zugeordnet werden kann und dieses gemäß einer Zielsetzung charakterisierend beschreibt.

Es wird deutlich, dass die Gewinnung, Haltung und Auswertung solcher Daten gerade im Hinblick auf die multiplen Kommunikationskanäle heutiger Geschäftsbeziehungen, z. B. persönlich (Ladengeschäft, Vertreterbesuch), telefonisch (Telefonverkäufer, Call Center), oder elektronisch (WWW, e-Mail), nur mithilfe von Informationssystemen effizient möglich ist. Im Anwendungsumfeld e-Business erscheint sogar eine enge Integration von Profiling-Mechanismen und den die Kerngeschäftsprozesse abbildenden Softwaresystemen notwendig.

Unter **Profiling** ist somit die Zusammenfassung der zur Gewinnung, Haltung und Auswertung von Profilen erforderlichen Prozesse zu verstehen.

Die Eigenschaft von Profiling, als Basis für verschiedene Anwendungen genutzt zu werden, drückt [Best00] prägnant aus: "Profiling is an enabling technology for personalization and much more..."

#### 1.1.5 Ziel der Diplomarbeit

Das Ziel dieser Diplomarbeit ist es, ein universelles Modell für das Profiling von Nutzern von e-Business-Websites zu erstellen, welches grundlegende Anwendungsfälle und daran beteiligte Objekte identifiziert sowie sinnvolle Vorgehensweisen und erforderliche Mechanismen zur automatisierten Erhebung, strukturierten Speicherung und Auswertung von Profildaten vorschlägt.

Dieses Referenzmodell soll so umfassend und variabel sein, dass sich durch Adaption und Variation für möglichst viele Geschäftsmodelle passende Instanzen daraus ableiten lassen, die als Implementierungsmaßgabe dienen können.

Zur Validierung von objektorientierter Analyse und Entwurf folgt die Skizzierung einer fachlichen und systemtechnischen Integration des erstellten Modells in zwei beispielhafte Internetanwendungen.

Die dadurch gewonnenen Erkenntnisse fließen in eine abschließende Bewertung ein und es erfolgt ein Ausblick auf eine mögliche Realisierung.

## **1.2 Kontext der Diplomarbeit**

Zur Einordnung des Themas dieser Arbeit folgt eine Vorstellung zugrunde liegender Literatur und vorhandener Softwareprodukte, die der Gewinnung, Haltung und Auswertung von Profildaten bei Websites dienen, um den aktuellen Stand der Wissenschaft und kommerziellen Anwendung darzustellen. Abschließend werden verwandte Techniken kurz erörtert, um eine semantische Abgrenzung des Begriffs 'Profiling' zu erreichen.

### 1.2.1 Literatur

Das spezielle Themengebiet 'Profiling von Nutzern von e-Business-Websites' wird nur in geringem Maße innerhalb der Fachliteratur behandelt. Einige der aktuellen Buchveröffentlichungen zu den Themen e-Business bzw. e-Commerce und deren technischer Unterstützung befassen sich zwar partiell mit den Möglichkeiten der Personalisierung von Websites, gehen dabei jedoch kaum auf notwendige Eigenschaften der dazu erforderlichen Profile und Möglichkeiten zu deren Gewinnung ein.

Dagegen existieren einige Konferenzbeiträge sowie Veröffentlichungen in wissenschaftlichen Zeitschriften aus den letzten zwei bis drei Jahren, die sich zumindest mit einzelnen Aspekten der Gewinnung und Auswertung von Profilinformatoren und darauf basierenden Möglichkeiten befassen.

Die Grundidee der Nutzung individueller Kundendaten zur Erreichung wirtschaftlicher Ziele, die über das Halten klassischer Stammdaten hinausgeht, findet sich in der Betriebswirtschaftslehre beim Database Marketing, dessen Entwicklung u. a. in [LiHa94] beschrieben wird. Darauf aufbauend entwickelten Peppers und Rogers zu Beginn der 90er Jahre das One-to-One-Marketing, dessen Grundkonzepte sie in [PeRo96] darlegen.

Bei diesen Ansätzen des Marketing wurde der Wert einer gut strukturierten und gepflegten Kundendatenbank erkannt und weiterführend die Erstellung umfassenderer Konsumentenprofile propagiert, die zur gezielten und personalisierten Ansprache potenzieller Kunden genutzt werden sollen.

Die mit der wachsenden Verbreitung der Internetnutzung einhergehende Verlagerung von Geschäftsprozessen ins Web bildet schließlich eine Plattform, die sich in besonderer Weise für die individuelle Kommunikation und Geschäftsabwicklung von Unternehmen eignet. Diesem Umstand wird in [PRD99] Rechnung getragen, indem Möglichkeiten der Implementierung von One-to-One-Strategien unter Ausnutzung der informationstechnologischen Unterstützung von Austauschprozessen zwischen Unternehmen und Kunden aufgezeigt werden.

Auf Seiten der Informatik und in wissenschaftlicher Weise befassen sich Forscher aus dem Gebiet des Data Minings mit der Analyse des Nutzerverhaltens auf Websites. Seit 1999 findet im Rahmen der 'ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)' der 'Workshop on Web Usage Analysis and User Profiling' (WEBKDD) statt, dessen Beiträge u. a. in [MaSp00] veröffentlicht wurden.

Der Schwerpunkt eines Großteils dieser Arbeiten liegt jedoch auf der Analyse von Webserver-Logfiles, während die Generierung individueller und einzelnen Nutzern

zuordenbarer Profile sowie deren Nutzung zur Laufzeit einer Webapplikation kaum berücksichtigt wird.

So wird in [Baum00] ein Algorithmus vorgestellt, der Navigationsmuster aus Logfiles extrahiert und dabei Wissen über die Webanwendung einbezieht, indem vorgegebene Folgen von Seitenabrufen, die Struktur der Verlinkung und Hierarchisierungen des Contents besonders berücksichtigt werden, um den speziellen Anforderungen beim Einsatz bei e-Commerce-Websites zu genügen. Als Ergebnis werden verschiedenartige Navigationsmuster ausgegeben, die zur Evaluierung der Website herangezogen werden können, das Verhalten der Nutzer jedoch nur in seiner Gesamtheit beschreiben.

Dagegen verwendet [Moba00] Navigationsmuster zur Derivation sogenannter "aggregate usage profiles", die unterschiedliche Typen von Nutzern charakterisieren und für Empfehlungsmechanismen verwendet werden können, indem eine Einordnung neuer Nutzer in die gefundenen Stereotypklassen stattfindet. Die Bildung individueller Profile wird dabei nicht vorgenommen.

Weitere Fachartikel finden sich in den Publikationen der universellen Konferenzen wie AMCIS und ICSE sowie spezieller Veranstaltungen, z. B. der 'Bled Electronic Commerce Conference'. Dabei werden jedoch einerseits Aspekte der Analyse und statistischen Auswertung der Nutzung von Websites betrachtet und nur selten Probleme aufgegriffen, die für die automatisierte Generierung charakterisierender Profile von Interesse sind, andererseits liegt die Zielsetzung in vielen Fällen in der Schaffung intelligenter 'User Agents', welche den Nutzern das Auffinden relevanter Webinhalte durch Abgleich mit einem Interessensprofil ermöglichen sollen, und es wird kein Bezug zu einer wirtschaftlich geprägten Nutzung der vorgestellten Verfahren aufgebaut.

Eine der wenigen Ausnahme bildet [KRR00], wo der kombinierte Einsatz von neuronalen Netzen und Fuzzy-Logik für die Auswahl von Werbebannern im Rahmen einer One-to-One-Strategie vorgeschlagen wird, und [ChJe00] beschreibt die Nutzung von Fuzzy-Logik zur Segmentierung von Kunden auf einer e-Commerce-Website im Bereich des Autohandels. Beide Beiträge verdeutlichen die Nachteile einer scharfen Unterteilung von Konsumenten, wie sie bei der regelbasierten Personalisierung vorkommt, betrachten jedoch nicht oder nur peripher die Gewinnung von Informationen über die potenziellen Kunden.

Unter den wissenschaftlichen Zeitschriften befassen sich einige der Fachjournale von Special Interest Groups (SIG) der ACM vereinzelt mit Aspekten des Profilings; hier sind insbesondere die SIGKDD (Knowledge Discovery in Data, s. o.) sowie die SIGecom (Electronic Commerce) zu nennen, wobei einige Artikel zur Gewinnung von Anregungen und Erreichung eines Überblicks über das Themengebiet hilfreich waren, aus den bereits o. g. Gründen jedoch nicht direkt in diese Arbeit einfließen konnten.

Die Ausgabe No. 8, Vol. 43 der 'Communications of the ACM' von August 2000 bietet mit dem Schwerpunktthema 'Personalization' ebenfalls eine gute Übersicht über dieses Forschungsgebiet, wobei auch spezielle Aspekte des Profilings berührt werden. Referenzen auf verschiedene Artikel dieser Ausgabe finden sich in den jeweiligen Abschnitten, in denen darauf Bezug genommen wird.

### 1.2.2 Produkte

Durch die Hinwendung von Wirtschaftsunternehmen zur elektronisch unterstützten Geschäftsprozessabwicklung einschließlich des Verkaufs von Waren und Dienstleistungen per Internet und die damit verbundene Relevanz von Profiling und Personalisierung für den vertrieblichen Erfolg existiert eine Reihe von Softwareprodukten, die das Erstellen, Verwalten und Auswerten von Konsumentenprofilen unterstützt sowie z. T. für darauf basierende Maßnahmen wie die Personalisierung von Websites genutzt werden kann.

Einige der wichtigsten oder in diesem Zusammenhang interessanten Produkte werden nachfolgend kurz vorgestellt, um einen groben Überblick über die am Markt verfügbare Software zu erhalten.

- BroadVision One-to-One Enterprise

BroadVision One-to-One Enterprise ist eine Anwendungssuite, die den Betrieb hochgradig personalisierter Websites erlaubt. Kern der Applikation ist eine Serversoftware, die in Verbindung mit einem Webserver die individuelle Verarbeitung von Client-Anfragen erlaubt, wobei die dynamische Generierung von Antwortseiten auf vordefinierten Templates basiert und unter Auswertung vorgegebener Regeln geschieht.

Zur Administration des Systems werden verschiedene Werkzeuge verwendet, mit denen insbesondere die Personalisierungsregeln editiert werden können.

Es existieren Zusatzkomponenten zum System, die verschiedene Ausprägungen von Online-Shops und weitere Möglichkeiten der Geschäftsabwicklung realisieren sowie Schnittstellen zur Anbindung von Fremdsystemen wie Call Center-Applikationen und CRM-Software.

BroadVision begann früh mit der Entwicklung seines Systems und orientierte sich dabei stark an den Zielsetzungen des One-to-One-Marketings. Die Einführung von One-to-One Enterprise gilt jedoch als aufwändig und kostenintensiv, eine exakte Anpassung an das zugrunde liegende Geschäftsmodell erfordert Implementierungsarbeit, insbesondere da wenig Unterstützung für das Generieren von Profilinformatoren geboten wird [LHHL97], [Broa99].

- ATG Dynamo Personalization Server

Der ATG Dynamo Personalization Server ist eine Zusatzkomponente zum Applikationsserver der Art Technology Group. Er lässt sich nur mit diesem gemeinsam einsetzen und durch einen sogenannten Scenario Server sowie einen Commerce Server zu einem leistungsfähigen Online-Shopsystem erweitern.

Die Stärke dieser Anwendungssuite liegt in der Unterstützung der aktuellen J2EE-Standards zur Entwicklung eigener Softwarekomponenten, welche die individuell erforderlichen Geschäftsprozesse abbilden, dabei aber auch vorhandene Funktionalitäten der optional mitgelieferten Zusatzkomponenten nutzen können.

So erlaubt der Personalization Server das implizite Anlegen anonymer Profile zu Beginn eines Websitebesuchs, die nach einer Registrierung oder einem Login dem Nutzer zuordenbar gespeichert werden. Zusammen mit dem Scenario Server ist eine Ereignisverarbeitung möglich, mit der das momentane Navigationsverhalten eines Nutzers zur Erweiterung der flexibel zu definierenden Profildaten oder zur sofortigen Personalisierung von Inhalten genutzt werden kann.

Die Administration des Systems geschieht mithilfe des Dynamo Control Centers, das mit seiner grafischen Oberfläche wie BroadVision das komfortable Editieren von Geschäfts- und Personalisierungsregeln durch Anwender mit geringem technischen Know-how ermöglicht.

Auch diese Anwendung ist nicht ready-to-use, sondern bedarf der individuellen Anpassung und Eigenentwicklung, welche jedoch durch die Unterstützung zahlreicher Standards (XML, LDAP) und die Integration von Schnittstellen zu verschiedenen Fremdsystemen, z. B. zu diversen *Content-Management-Systemen* (CMS), erleichtert wird.

Der ATG Personalization Server ist jedoch nicht als eigenständige Anwendung für die Personalisierung einer bestehenden Webapplikation einsetzbar. Zudem kann die Anreicherung der Profile zwar sowohl explizit durch das Ausfüllen von Online-Formularen durch die Nutzer als auch implizit durch Beobachtung deren Navigationsverhaltens geschehen, im letztgenannten Fall findet jedoch nur die Protokollierung einer technischen Interaktion statt, der nicht ohne weiteres eine Semantik zugeordnet werden kann, die zur Charakterisierung der Nutzer beitragen würde.

Darüber hinaus existieren folgende weitere Produkte, die den bisher genannten ähneln oder Mechanismen zum Profiling zumindest in Ansätzen implementieren:

Broadbase dient zur Erstellung von Kundenprofilen auf Websites und Kontrolle darauf basierender Maßnahmen. Es eignet sich u. a. zum Einsatz in Kombination mit BroadVision und ATG Dynamo.

Der BEA Personalization Server ist Teil einer Anwendungssuite auf Basis des BEA Weblogic Applikationsservers und entspricht hinsichtlich der produktpolitischen Positionierung dem Personalization Server von ATG.

Siebel erstellt und vertreibt eine umfangreiche Produktfamilie von Customer Relationship Management-Anwendungen, die eine Vielzahl möglicher Kommunikations- und Vertriebskanäle abdeckt und dabei integrativ eine Personalisierung von Kommunikation und Interaktion mit Kunden erlaubt.

Somit ist es sinnvoll, Anforderungen an und Möglichkeiten des Profiling in Form eines Referenzmodells zu erarbeiten, da die vorgestellten Lösungen i. Allg. nicht universell einsetzbar sind, das Generieren und Verarbeiten von Profildaten nur peripher realisieren und dabei zahlreiche Schwachpunkte offenbaren, insbesondere hinsichtlich einer automatisierten Gewinnung von Profilinformatoren sowie deren semantischer Auswertung.

### 1.2.3 Anwendung

Die in Kapitel 1.1.4 vorgestellten Motive veranlassen schon heute viele Betreiber von Websites, Verhalten und Präferenzen ihrer Nutzer individuell zu speichern bzw. um personenbezogene Daten zu ergänzen. Insbesondere die bereits erwähnten Portale MSN und MyYahoo! erlauben umfassende Möglichkeiten der Personalisierung durch die Nutzer bis hin zum Editieren des eigenen Profils einschließlich Angabe persönlicher Informationen wie Interessen, Hobbys, e-Mail-Adresse und Foto, die daraufhin, sofern gewünscht, auch von anderen Nutzern eingesehen werden können.

Art und Umfang der beim Anbieter gespeicherten Daten unterliegen dabei der Zielsetzung der jeweiligen Website: der Internet-Buchhändler Amazon speichert zur Vereinfachung des Bestellvorgangs die von einem Kunden bevorzugten Versand- und Zahlungsinformationen, welche erst durch die Kombination mit den Transaktionsdaten der verkauften Artikel zu charakterisierenden Profilen werden, die daraufhin zu weiteren Produktempfehlungen genutzt werden können; beim Versicherungsmakler Quinner hinterlegen Interessenten über das Internet umfangreiche Angaben zu Beruf, Einkommen, Besitzverhältnissen u. ä., die einerseits für das Zusammenstellen eines Versicherungspakets erforderlich sind, andererseits den Nutzer soziodemografisch exakt beschreiben und durch die Sensibilität der Daten ein enormes Missbrauchspotenzial eröffnen.

Neben dem Einsatz von Profilen, der auch im Interesse der Nutzer liegt, besteht der Anreiz, durch die Erhebung charakterisierender Informationen Profile zu erstellen, deren Nutzung den Konsumenten keinen unmittelbaren Vorteil verschafft oder widerstrebt. Das Unternehmen DoubleClick setzt als weltweit größter Vermarkter von Online-Werbung umfangreiche technische Maßnahmen ein, um das Surfverhalten einzelner Nutzer zu protokollieren, damit einerseits Werbebanner möglichst zielgruppengenau eingeblendet werden können, andererseits ein Banner-Motiv demselben Nutzer nicht wiederholt gezeigt wird.

Insbesondere die Verknüpfung von Konsumentenadressen mit Daten über deren Surf- und Kaufgewohnheiten sind für die Werbewirtschaft von großem Interesse, derartige Profile werden vor allem in den USA intensiv und zu hohen Preisen gehandelt.

### 1.2.4 Verwandte Methoden und Techniken

Abschließend werden mit dem Profiling verwandte und daran angrenzende Methoden und Techniken kurz erörtert, um bestehende Unterschiede hervorzuheben und eine Abgrenzung zu schaffen.

- Bei der Logfile-Analyse handelt es sich um die Auswertung der von einem Webserver geschriebenen Protokolldateien. Diese enthalten die Zeitpunkte aller Seitenabrufe, die IP-Nummern oder Namen der Client-Rechner, die exakten URLs der Requests sowie einige weitere Informationen über die Systeme der Nutzer.

Als Beispiel wird nachfolgend ein Ausschnitt aus dem Zugriffs-Logfile des Webservers der adesso AG ([www.adesso.de](http://www.adesso.de)) vom 1. August 2000 wiedergegeben. Der Übersichtlichkeit halber wurden Trennstriche zwischen den einzelnen Requests eingefügt.

```
194.175.175.2 - - [01/Aug/2000:08:18:50 +0200] "GET
/de/wir_ueber_uns/das_unternehmen/Das_Unternehmen.html HTTP/1.0" 200 7906
"http://www.adesso.de/de/Homepage.html \->
/de/wir_ueber_uns/das_unternehmen/Das_Unternehmen.html" "Mozilla/4.0
(compatible; MSIE 5.01; Windows NT)"

194.175.175.2 - - [01/Aug/2000:08:18:50 +0200] "GET /media/images/adesso-
logo.gif HTTP/1.0" 200 4098
"http://www.adesso.de/de/wir_ueber_uns/das_unternehmen/Das_Unternehmen.html
```



```

\-> /media/images/ adesso-logo.gif" "Mozilla/4.0 (compatible; MSIE 5.01;
Windows NT)"

194.175.175.2 - - [01/Aug/2000:08:19:25 +0200] "GET
/de/wir_ueber_ uns/partnerschaften/Partnerschaften.html HTTP/1.0" 200 9349
"http://www. adesso.de/de/wir_ueber_ uns/das_unternehmen/Das_Unternehmen.html
\-> /de/wir_ueber_ uns/partnerschaften/Partnerschaften.html" "Mozilla/4.0
(compatible; MSIE 5.01; Windows NT)"

194.175.175.2 - - [01/Aug/2000:08:19:25 +0200] "GET /media/images/morse-
logo.gif HTTP/1.0" 200 4098
"http://www. adesso.de/de/wir_ueber_ uns/partnerschaften/Partnerschaften.html
\-> /media/images/morse-logo.gif" "Mozilla/4.0 (compatible; MSIE 5.01;
Windows NT)"

194.175.175.2 - - [01/Aug/2000:08:19:25 +0200] "GET /media/images/unido-
logo.gif HTTP/1.0" 200 4098
"http://www. adesso.de/de/wir_ueber_ uns/partnerschaften/Partnerschaften.html
\-> /media/images/unido-logo.gif" "Mozilla/4.0 (compatible; MSIE 5.01;
Windows NT)"

195.135.146.180 - - [01/Aug/2000:08:20:43 +0200] "GET / HTTP/1.0" 301 241 "-
\-> /" "Mozilla/4.7 [de] (WinNT; I)"

195.135.146.180 - - [01/Aug/2000:08:20:43 +0200] "GET /de/Homepage.html
HTTP/1.0" 200 9725 "- \-> /de/Homepage.html" "Mozilla/4.7 [de] (WinNT; I)"

195.135.146.180 - - [01/Aug/2000:08:20:44 +0200] "GET
/media/images/header1.jpg HTTP/1.0" 200 8196
"http://www. adesso.de/de/Homepage.html \-> /media/images/header1.jpg"
"Mozilla/4.7 [de] (WinNT; I)"

```

Zunächst sind zwei Seitenabrufe eines Clients mit der IP-Nummer 194.175.175.2 zu erkennen: um 8:18:50 h wurde eine Seite mit der Unternehmensbeschreibung übertragen ('Das\_Unternehmen.html'), die über die Startseite der Website ('Homepage.html') erreicht wurde; 35 Sekunden später navigierte der Nutzer von dort aus zu einer Beschreibung der Unternehmenspartnerschaften ('Partnerschaften.html'). Diese beiden *Pageviews* haben insgesamt fünf HTTP-Requests ausgelöst (*Hits*), da jeweils Grafiken in den abgerufenen HTML-Dateien eingebettet waren, welche einzeln nachgeladen werden.

Um 8:20:43 h besuchte ein weiterer Nutzer (IP-Nummer 195.135.146.180) die Website durch Eingabe von 'www. adesso.de' in seinen Browser, einem Netscape Navigator in Version 4.7 ('Mozilla/4.7'), bei dem Deutsch als bevorzugte Sprache eingestellt war.

Die vorherigen Requests wurden hingegen von einem Microsoft Internet Explorer 5.01 ausgelöst ('Mozilla/4.0 (compatible; MSIE 5.01; Windows NT)') und beide Nutzer verwendeten Windows NT als Betriebssystem auf ihren Rechnern.

Typische Werkzeuge zur Logfile-Analyse, wie beispielsweise Exody WebSuxess, veranschaulichen die Anzahl der Seitenabrufe und Websitebesuche innerhalb einer Zeitspanne, liefern Aufschlüsselungen der am häufigsten und seltensten besuchten Seiten, der von den Nutzern verwendeten Browser und Betriebssysteme und nennen u. U. typische Abfolgen von Seitenabrufen. Hierdurch wird ein Rückblick auf die Nutzung einer Website durch die Gesamtheit aller Nutzer möglich, es kann i. d. R. keine Unterscheidung von individuellen Besuchern oder eine Verknüpfung von deren Verhalten mit zusätzlichen Informationen vorgenommen werden.

- Methoden des Data Minings, das in diesem Zusammenhang oft Web Mining genannt wird, versuchen, Wissen über den Inhalt, die Struktur oder die Nutzung des Internets

aus Quell- und Zugangsdaten zu extrahieren [Walt01]. Im letztgenannten Fall basieren sie meist ebenfalls auf der Auswertung der Webserver-Logfiles, bedienen sich jedoch aufwändigerer Analysemethoden, indem sie wie in [Spil00] Navigationsmuster aus Logfiles rekonstruieren, aufbereiten, teilweise visualisieren und auf typische Pfade hin untersuchen. Die aufbereiteten Daten eignen sich auch zur Clusteranalyse oder Anwendung von Methoden der Künstlichen Intelligenz, mit denen beispielsweise eine Vorhersage des Navigationsverhaltens oder bestimmter Nutzereigenschaften versucht wird. Die Ergebnisse derartiger Analysen können für die Optimierung einer Website eingesetzt werden, die wie in [PeEt00] gezeigt sogar teilweise automatisiert erfolgen kann.

Aufgrund der aufwändigen Datenaufbereitung und -analyse eignen sich diese Verfahren nicht zur Personalisierung zur Laufzeit und berücksichtigen zumeist nicht die Individualität einzelner Nutzer sowie möglicherweise vorhandene Zusatzdaten.

- Das Collaborative Filtering ist eine der Methoden des Data Minings und nutzt die anhand verschiedener Metriken ermittelte Ähnlichkeit im Navigations- oder Transaktionsverhalten von Websitenutzern bzw. deren explizite Bewertungen von Inhalten oder Produkten [MAB00]. Damit ist Collaborative Filtering neben Content Based Filtering für die Realisierung von Empfehlungsmechanismen geeignet, indem einem Nutzer die von ähnlich eingestufteten Nutzern präferierten Inhalte zur Ansicht bzw. Produkte zum Kauf angeboten werden; beispielsweise beruhen die beim Internet-Buchhändler Amazon gegebenen Kaufempfehlungen auf diesem Prinzip und die Firma Net Perceptions bietet entsprechende Softwareprodukte an.
- Das Content Based Filtering basiert im Gegensatz dazu auf einer errechneten Ähnlichkeit zwischen klassifizierten oder beschriebenen Inhalten bzw. Produkten und von Nutzern angegebenen Präferenzen [MCS00]. Zur Klassifizierung von Webinhalten werden dabei häufig eingebettete Meta-Tags oder Volltextsuchmaschinen eingesetzt, wodurch jedoch z. T. unbefriedigende Ergebnisse erzielt werden, da Treffer nur bei syntaktischer Übereinstimmung von Zeichenketten auftreten und Empfehlungen am Rande des Interessenprofils kaum möglich sind. Abhilfe schaffen ein kombinierter Einsatz des Content Based Filtering mit dem Collaborative Filtering und die Einbeziehung von Bewertungen der ermittelten Empfehlungen durch den Nutzer.
- Regelbasierte Personalisierung bezeichnet die Vorabdefinition von überwiegend *if-then*-artigen Anweisungen zur Zusammenstellung der auf Webseiten dargebotenen Inhalte aufgrund vermuteter Zusammenhänge zwischen bekannten Eigenschaften eines Nutzers und dessen möglichen Interessen. Das in Kapitel 1.2.2 vorgestellte BroadVision One-to-One Enterprise ist das bekannteste Produkt, welches dieses Verfahren verwendet [MCS00].  
Bei der Aufstellung der erforderlichen Regeln werden oftmals Erkenntnisse aus Logfile-Analysen oder des Data Minings einbezogen.

Profiling als das automatisierte Gewinnen, Halten und Auswerten individueller charakterisierender Informationen kann somit die Verwendung von Webserver-Logfiles überflüssig machen, da die erstellten Profile als Basis sowohl für einfache Analysen als auch zur Anwendung von Methoden des Data Minings dienen können. Je nach Art und Umfang

der gespeicherten Profildaten ist dabei nicht nur die Verwendung von Navigations- und Transaktionsdaten, sondern auch die Einbeziehung soziodemografischer Daten wie Geschlecht, Alter, Familienstand usw. möglich, was zu einer Verbesserung der Ergebnisse führen sollte. Die daraus gewonnenen Erkenntnisse können wiederum in den einzelnen Profilen gespeichert werden, um zusätzlich zu den vorhandenen Informationen beispielsweise zur Laufzeit-Personalisierung von Websites mittels regelbasierter Systeme verwendet zu werden, welche möglichst umfassender Profildaten bedürfen.

## **2 Definition von Anforderungen an Profiling**

In diesem Kapitel werden Anforderungen an Profiling<sup>3</sup> herausgearbeitet, welche nach den Phasen Gewinnung, Haltung und Auswertung von Profilinginformationen gegliedert sind.

Diese Anforderungen befassen sich mit breit gestreuten Aspekten des Profilings und werden im Sinne des Referenzcharakters des zu erstellenden Modells isoliert und möglichst neutral, d. h. ohne Blick auf eine bestimmte spätere Umsetzung oder Anwendung formuliert, was zu Zielkonflikten führen oder über die Anforderungen eines konkreten Anwendungsfalls weit hinaus reichen kann.

### **2.1 Anforderungen an die Profildatengewinnung**

Grundlegende Überlegungen zur Gewinnung von Profildaten betreffen die Art der Nutzererfassung und der Generierung von Daten, Integrationsmöglichkeiten für Profildaten anderer Kommunikationskanäle, aber auch Aspekte der Wahrung der Privatsphäre und des Datenschutzes, da diese bereits bei der Erhebung von profilrelevanten Informationen beachtet werden müssen.

#### **2.1.1 Nutzererfassung**

Unter Nutzererfassung soll verstanden werden, ob und in welcher Weise eine Identifizierung und Deanonymisierung einzelner Besucher von Websites erfolgen soll.

##### **2.1.1.1 Anonyme Nutzer**

Ein System zur Gewinnung von Nutzerprofilen muss in der Lage sein, anonyme Nutzer zu handhaben, Profile zu diesen aufzubauen und zu verwalten.

Diese Anforderung ergibt sich daraus, dass das frei zugängliche Angebot von Informationen auf Websites der Regelfall ist und Nutzer nicht gezwungen werden, sich zu registrieren und mit der dabei vergebenen Kennung am System anzumelden. Als einzig praktikable Möglichkeit trifft dies für Websites zu, die eine hohe Anzahl häufig wiederkehrender Besucher erreichen wollen, welche von der Unbequemlichkeit eines Anmeldeprozesses abgeschreckt würden, sowie für Websites, deren Informations- oder Leistungsangebot keinen ausreichenden Anreiz für Nutzer bietet, diese Mühe auf sich zu nehmen.

Dennoch ist es auch in solchen Fällen interessant, das Navigations- und Interaktionsverhalten der Besucher in individuellen, anonymen Profilen zu speichern, um darauf basierend beispielsweise das Informationsangebot verbessern oder Online-Werbung personalisiert platzieren zu können.

Weiterhin sollte versucht werden, derart erfasste Nutzer bei späteren Website-Besuchen wiederzuerkennen und das bereits angelegte Profil zu erweitern, sofern diese Identifikation technisch möglich ist.

Dies ist sinnvoll, da ein zunächst anonymer Nutzer beispielsweise beim Abschluss einer Online-Warenbestellung zur Bekanntgabe persönlicher Daten wie Name und Anschrift

---

<sup>3</sup> Die nachfolgend aufgestellten 'Anforderungen an (ein Modell für) Profiling' sind letztendlich funktionale und nicht-funktionale Anforderungen an ein Softwaresystem, das auf diesem Modell basiert, deren Berücksichtigung bei Modellierung und Entwurf jedoch besonders wichtig ist.

gezwungen wird oder durch Vergabe einer Kundennummer ein eindeutig kennzeichnendes Attribut erhält, was eine Identifizierung der Person bei zukünftigen Transaktionen erleichtert. Auf diese Weise können die bereits vorher erhobenen, eventuell umfangreichen Profildaten weiter verwendet werden.

### 2.1.1.2 Registrierte Nutzer

Anbieter, deren Online-Angebote einen von Nutzern als hoch empfundenen Wert haben, können mit deren Akzeptanz rechnen, wenn diese Inhalte nur nach Registrierung und Anmeldung bei der Website zugänglich gemacht werden. Hierzu zählen u. a. die zahlreichen Anbieter von kostenlosen Dienstleistungen wie Webpace, Fax-, SMS- und e-Mail-Versand, sowie Unternehmen, die Software oder Multimedia-Inhalte zum freien Download anbieten [Sto199].

Die Identifikation einzelner Nutzer ist dadurch sichergestellt, dass diese sich zu Beginn jeder *Session* unter Angabe einer eindeutigen Kennung und geschützt durch ein Passwort an der serverseitigen Applikation anmelden, was unmittelbar für die Zuordnung und Fortführung eines korrespondierenden Profils genutzt werden sollte.

Eine solche Registrierung muss nicht zwangsläufig die Nennung von Name und Anschrift des Nutzers erfordern, was häufig zur Durchführung von Offline-Marketingmaßnahmen erwünscht ist; als Kompromiss kann eine Identifizierung der Nutzer über deren Kennung erfolgen, die damit als Synonym fungiert, ohne dass der echte Name der betreffenden Person bekannt ist oder damit in Verbindung gebracht werden kann.

### 2.1.2 Einzel- und Gruppenprofile

Eine weitere Basisanforderung ist es, die Zugehörigkeit einzelner Nutzer zu verschiedenen, frei definierbaren Gruppen zu verwalten sowie Profile, die Eigenschaften solcher Gruppen repräsentieren, zu pflegen.

Dadurch soll es einerseits möglich sein, die Mitgliedschaft in Communities, geschlossenen Nutzergruppen oder die Zugehörigkeit zu Organisationen innerhalb der Nutzerprofile auszudrücken, andererseits können auf diese Weise aggregierte Profilinformatoren einer gesamten Gruppe für auf Profiling basierende Maßnahmen verwendet werden.

Die zu unterscheidenden Gruppen können dabei Organisationsstrukturen der realen Welt entsprechen oder aufgrund interner Anforderungen der Webapplikation oder des Profiling-Systems definiert werden.

Auf vielen Websites können sich Nutzer beispielsweise in Gruppen strukturieren (groups.yahoo.com, www.c-lounge.com) oder werden aufgrund ihrer Eigenschaften unterschiedlichen Berechtigungsgruppen zugeordnet (Laie ⇔ Experte im Online-Chat, Anbieter ⇔ Nachfrager auf Auktions-Websites).

Die Kombination von Einzel- und Gruppenprofilen erlaubt z. B. auf einer B2B-Website für e-Procurement die Handhabung eines Teams von Einkäufern, die für das gleiche Unternehmen arbeiten. Jedes Teammitglied kann sowohl individuell angesprochen werden und persönliche Präferenzen und Daten hinsichtlich der Systembenutzung hinterlegen, alle Teammitglieder verwenden jedoch auch einen Teil gemeinsamer Basisdaten, beispielsweise Zahlungs- und Versandinformationen, eine gemeinsame Kreditlinie, offene Posten usw., die im Gruppenprofil hinterlegt werden können.

Schließlich erlaubt ein solches Gruppenkonzept auch die systeminterne Zusammenfassung von Nutzern mit ähnlichen Profilen sowie den Vergleich von Gruppenprofilen.

### 2.1.3 Profildatengewinnung

Bei der Art der Gewinnung von Profildaten kann grundlegend zwischen expliziten und impliziten Vorgehensweisen unterschieden werden. Diese bringen jeweils Vor- und Nachteile mit sich, die eine Kombination der beiden Möglichkeiten als besonders sinnvoll erscheinen lassen.

#### 2.1.3.1 Explizite Profildatengewinnung

Als explizite Gewinnung von Profildaten wird die von Nutzern freiwillig und selbstständig durchgeführte Preisgabe profilrelevanter Informationen bezeichnet.

Dies wird zurzeit überwiegend durch Ausfüllen von Online-Fragebögen erreicht, wofür Internetnutzern verschiedene Anreize geboten werden können. Die o. g. Registrierung zur Nutzung kostenloser Dienstleistungen wird häufig für die Erhebung soziodemografischer Daten wie Alter, Geschlecht, Familienstand, Beruf, Bildungsgrad und Einkommen genutzt, aber auch Hobbys, Interessensgebiete oder die Hard- und Softwareausstattung des verwendeten Computers werden erfragt.

Eine weiterreichende Maßnahme ist es, Besuchern von Websites das freie Editieren ihres eigenen Profils zu ermöglichen. Vorteil dieser Vorgehensweise ist, dass die Akzeptanz des Profilings an sich gesteigert werden kann, da die Nutzer ein hohes Maß an Kontrolle über ihre Profildaten erhalten. In Verbindung mit einer nutzerfreundlichen Privacy-Politik (s. u.) kann dies zu einer positiv gewerteten Abgrenzung gegenüber aggressiv Daten sammelnden Unternehmen führen und zudem das Maß bewusster Falschangaben reduzieren.

Schließlich eröffnet die freie Editierbarkeit eigener Profile die Chance, wertvolle Informationen über Nutzer zu erfahren, die bei Anwendung der *Clickstream*-Analyse (s. u.) gar nicht, nur schwer oder erst zu einem späteren Zeitpunkt hätten gefolgert werden können. Voraussetzung ist jedoch, dass die auf den Profildaten basierende individuelle Ausgestaltung der Geschäftsbeziehung einen echten Mehrwert für den potenziellen Kunden mit sich bringt, beispielsweise in Form von Zeitersparnis, besonders hochwertigen Informationen, limitierten Vorzugspreisen oder exklusiven Clubangeboten.

#### 2.1.3.2 Implizite Profildatengewinnung

Als implizite Gewinnung von Profildaten ist die Beobachtung und Auswertung von Interaktionen zu verstehen, an denen ein Nutzer bzw. Kunde im Rahmen der gewöhnlichen Geschäftsbeziehung beteiligt ist, ohne dabei unmittelbaren Einfluss auf die Profildatengewinnung zu nehmen.

##### 2.1.3.2.1 Clickstream

Beim Kommunikationskanal WWW ist hierunter die Protokollierung der Nutzung einer Website gemeint, welche i. d. R. auf der Erfassung einzelner Requests basiert. Damit ist die Navigation eines Nutzers von Link zu Link, der sogenannte Clickstream, auf Seite des Servers rekonstruierbar. Die Möglichkeit der Aufzeichnung und Analyse dieser Daten ist eine

Grundanforderung an ein System für Profiling und muss deshalb bei der Modellbildung berücksichtigt werden.

Da das Protokollieren einzelner HTTP-Requests zur Standardfunktionalität von Webservern gehört, existieren bereits zahlreiche Werkzeuge, welche die Auswertung dieser Logfiles unterstützen. Dabei stehen oft folgende Fragen im Mittelpunkt:

- Welche sind die am häufigsten/am seltensten besuchten Seiten?
- Welche sind die häufigsten Einstiegs-/Ausstiegsseiten einzelner Besuche (Sessions)?
- Wie lange verweilen Besucher auf einzelnen Seiten und insgesamt?
- Wie viele Seiten werden pro Session angeschaut (Pageviews pro Session)?
- Welche typischen Navigationspfade gibt es?
- Über welchen Link wurde die Website erreicht (Referer)? Handelt es sich dabei eventuell um eine Suchmaschine oder ein Werbebanner, und falls ja, welche Suchbegriffe wurden eingegeben bzw. auf welcher Website war welches Banner geschaltet?
- Aus welchen Ländern und über welche Netze kommen die Nutzer?
- Welche Betriebssysteme und Browser verwenden die Nutzer?

Im Gegensatz zur Logfile-Analyse, die in größeren Zeitabständen angewendet wird und statistische Aussagen über die Gesamtheit der Nutzer einer Website erlaubt, muss ein Profilingssystem in der Lage sein, die gleichen Daten den einzelnen Nutzern zuordenbar zu erfassen und dahin gehend weiterzuverarbeiten, dass eine automatische Erweiterung des korrespondierenden Nutzerprofils auf semantischer Ebene erfolgt.

Dies bedeutet beispielsweise, dass der Click eines Surfers auf einen Link zum Abruf einer Datei "prod\_heim\_u\_garten.html" zu einem Profileintrag führen sollte, der das Interesse des Nutzers an Produkten eines Anbieters aus dem Bereich "Heim und Garten" eindeutig und zeitnah auswertbar repräsentiert. Dieses Zuordnungsproblem ist weder fachlich noch technisch trivial und wird in Kap. 3.4 näher diskutiert.

Darüber hinaus sollten wie bei Logfiles alle weiteren mittels HTTP erkennbaren Daten über den Client gespeichert werden, da hieraus Profilinformatoren generiert werden können bzw. die Identifikation eines Nutzers erleichtert wird. So stellt die Betriebssystem- und Browserversion eines Nutzers eine wichtige Information für Anbieter von Software dar, und exakte Zeitstempel zu jedem Click können zur Ermittlung von Surfgehnheiten oder Gewichtung von Interessen bei Seitenabrufen herangezogen werden.

### 2.1.3.2.2 Transaktionen

Neben einfachen Seitenabrufen, die quantitativ sicherlich die größte Rolle bei der Nutzung von Websites spielen, sind insbesondere geschäftliche Transaktionen Ziel eines umfassenden Profiling im e-Business. Verbindliche Beantragungen, Bestellungen oder direkte Online-Verkäufe von Waren oder Dienstleistungen sind letztendlich das Ziel unternehmerischer Tätigkeit und innerhalb der elektronischen Geschäftsabwicklung die primäre Messgröße für dessen Erfolg.

Der Begriff der **geschäftlichen Transaktion** soll dabei so weit gefasst verstanden werden, dass jegliche Kommunikation inbegriffen ist, die Einfluss auf die Abwicklung von Geschäftsprozessen hat bzw. wirtschaftliche oder rechtliche Relevanz besitzt.

Dies schließt beispielsweise mithilfe elektronischer Medien durchgeführte Änderungen oder Stornierungen von Warenbestellungen, Schadensmeldungen und -abwicklungen bei Versicherungen, Beantragungen amtlicher Unterlagen und Vorgänge oder Meldungen von Zählerständen an Versorgungsunternehmen durch Verbraucher ein.

Die separate Erfassung von Online-Transaktionsdaten ergibt sich einerseits aus der Notwendigkeit der Integration mit Daten aus dem Offline-Geschäft (s. Kapitel 2.1.4), andererseits bildet sie eine wertvolle Grundlage für die elektronisch unterstützte Personalisierung von Geschäftsbeziehungen, da mit ihr z. B. automatisiert Treuerabatte kalkuliert oder Empfehlungen für Wiederbeschaffungskäufe bei Verbrauchsgütern generiert werden können.

### 2.1.3.2.3 Weitere Interaktionen

Abschließend werden alle weiteren Interaktionen zusammengefasst, die zwar keine wirtschaftlich bedeutsame Transaktion darstellen, jedoch über den einfachen Abruf einer HTML-Seite hinausgehen. Dazu zählen insbesondere:

- Eingaben in Suchmaschinen  
Die Eingabe eines Suchbegriffs stellt ein starkes Indiz für ein vorhandenes Interesse eines Nutzers dar. Darüber hinaus kann bei der Analyse der Suchhäufigkeit bestimmter Begriffe auf Verbesserungspotenzial hinsichtlich Inhalt oder Struktur der Website geschlossen werden.
- Download von Dateien  
Der Download von Dateien deutet ebenfalls auf ein überproportional zu gewichtendes Interesse eines Nutzers an deren Inhalt an, welches zur Profilbildung genutzt werden sollte.
- Votings  
Elektronische Abstimmungen bieten die Möglichkeit, eindeutige Meinungsäußerungen eines Nutzers zu konkreten Fragestellungen zu erhalten.
- Chat, Postings und Mailversand  
In gleicher Weise sollte die Beteiligung an Chats und Bulletin-Board Systemen sowie die Benutzung von e-Mail-Versandfunktionen, wie sie z. B. zur Weiterempfehlung redaktioneller Artikel dienen, protokolliert werden, sofern sich aus der Interaktion des Nutzers mit ausreichend hoher Sicherheit ein bestimmtes Interesse schließen lässt.

### 2.1.4 Integrationsmöglichkeit für Profilinformatoren anderer Kommunikationskanäle

Eine wichtige Anforderung an ein Profilingssystem ist die Möglichkeit zur Integration von Profilinformatoren, die an anderen Kontaktstellen der Anbieter-Kunde-Beziehung anfallen. Diese Notwendigkeit ergibt sich umso mehr, da kundenorientierte Unternehmen heutzutage multiple Kommunikationskanäle anbieten. Dabei kann es sich sowohl um elektronisch unterstützte als auch um klassische Kommunikationsmittel handeln; nachfolgend sind die wichtigsten genannt:



- Persönlicher Kontakt: in Geschäfts- oder Verkaufsstellen eines Unternehmens (brick-and-mortar), durch Vertreterbesuch, auf Messen
- Post: in Form von Direktwerbung (Mailings), zur Abwicklung von Transaktionen (z. B. mit Versicherungen, Versandhandel)
- Klassische Telekommunikation: Telefon (Call Center, Telefonverkauf, Telefoncomputer) und Telefax
- Moderne Telekommunikation: EDI, e-Mail, SMS, WAP

Die Bedeutung der Zusammenführung von Daten, die eine ganzheitliche Sicht auf Kunden und Interessenten ermöglicht, ergibt sich daraus, dass ein einzelner Kunde im Regelfall über mehr als einen Kanal mit einem Unternehmen kommuniziert und Produkte bezieht bzw. Leistungen nachfragt. Beispielsweise sollte das Potenzial eines im Offline-Vertrieb umsatzstarken Versandhauskunden auch nach Identifikation bei dessen erstem Website-Besuch erkannt werden, und vorhandene Daten, beispielsweise über bereits gekaufte Produkte oder Präferenzen, für eine mögliche Personalisierung genutzt werden.

Dies impliziert die Notwendigkeit des Datenaustauschs eines Profiling-Systems mit bestehenden Kundendatenbanken oder Warenwirtschaftssystemen; im Idealfall einer streng marktorientierten Organisation wäre eine einzige, umfassende Kundendatenbank einschließlich darin enthaltener Profilinginformationen zentraler Bestandteil der informationstechnologischen Infrastruktur.

### 2.1.5 Einfache und individuelle Anpassbarkeit der Profiling-Intensität

Art und Umfang der zu sammelnden Profilinginformationen sollten bei einem entsprechenden System einfach und individuell anpassbar sein. Um den im nachfolgenden Abschnitt beschriebenen Anforderungen an den Schutz der Privatsphäre und des Datenschutzes gerecht zu werden, sollten wesentliche Aspekte der Profildatengewinnung, -haltung und -auswertung, z. B. die Anonymisierung von Profilen oder das Aufzeichnen des Clickstreams, global und zur Laufzeit des Systems konfigurierbar sein, um eine einfache Anpassung beispielsweise an rechtliche Rahmenbedingungen zu gewährleisten.

Darüber hinaus muss innerhalb der globalen Einstellungen eine weiterreichende individuelle Anpassung des Profiling möglich sein, um den Bedürfnissen einzelner Nutzer gerecht zu werden. So wie zur Gewinnung zusätzlicher Informationen den Nutzern das zumindest teilweise Editieren des eigenen Profils ermöglicht werden sollte, kann zur Steigerung der Akzeptanz der Profilingmaßnahmen eine individuelle Vereinbarung über die Intensität der Datensammlung und –verarbeitung getroffen werden. Nutzer sollten beispielsweise festlegen können, ob sie e-Mail-Werbung eines Anbieters empfangen möchten oder der Weitergabe persönlicher Daten an Dritte zustimmen; derartige Einwilligungen sind in Deutschland aus juristischen Gründen in vielen Fällen ohnehin erforderlich.

### 2.1.6 Privatsphäre und Datenschutz

Weitere zentrale Anforderungen, die in einem Referenzmodell für Profiling bei e-Business-Websites berücksichtigt werden müssen, sind der Schutz der Privatsphäre der Nutzer, häufig

englisch als 'Privacy' bezeichnet, sowie die Einhaltung datenschutzrechtlicher Bestimmungen.

Bereits in [PeRo96] wird als eine der Handlungsmaximen des One-to-One-Marketings gefordert, "die Privatsphäre des Kunden (zu) beschützen, nicht (zu) bedrohen!"

Der Begriff Privatsphäre wird dabei in Anlehnung an vorangegangene Arbeiten zweigeteilt definiert [PeRo96]:

1. Kontrolle über die Verbreitung und Nutzung von Informationen über eine Person durch die Person selbst.
2. Kontrolle über das Eindringen anderer Personen in das individuelle Leben einer Person.

Der Grundsatz des Schutzes der Privatsphäre wird in vielen auf Peppers und Rogers aufbauenden Marketing-Konzepten und Geschäftsmodellen aufrecht erhalten, da diese ein hohes Maß an Loyalität und Kundenzufriedenheit zum Ziel haben.

Die Wichtigkeit dieser Forderung belegen aktuelle Trends aus USA, wo es nach [Volo00] im Vergleich zur europäischen Union und insbesondere zu Deutschland nur wenige und sehr liberale gesetzliche Bestimmungen zum Datenschutz von Privatpersonen gibt, die Öffentlichkeit jedoch, angeführt von Verbraucherschutzorganisationen, in hohem Maße sensibilisiert den Aktivitäten Daten sammelnder Unternehmen gegenüber steht.

Die Einhaltung gesetzlicher Datenschutzbestimmungen ist unbedingt erforderlich, aber auch freiwillig auferlegte Beschränkungen zum Schutz der Privatsphäre sind zu beachten, um die Glaubwürdigkeit eines Anbieters nicht zu gefährden. So stellt die Verknüpfung von Nutzungs- und Personendaten in Deutschland einen Verstoß gegen das Teledienstedatenschutzgesetz (TDDSG) dar, sofern der einzelne Nutzer dieses Vorgehen nicht ausdrücklich gestattet hat.

Hieraus ergibt sich die Anforderung, Privacy-Einstellungen individuell zu verwalten und die Profildatengenerierung und –auswertung spezifisch daran auszurichten, sowie die systemweite Konfiguration wichtiger globaler Datenschutzaspekte möglichst einfach, beispielsweise per Parameter, zu ermöglichen.

## 2.2 Anforderungen an die Profildatenhaltung

Eng verknüpft mit den Aspekten der Profildatengewinnung ist die Frage nach der Art der Datenhaltung, insbesondere hinsichtlich Erweiterbarkeit, Strukturierung und Werteskalierung der zu sammelnden Informationen. Darüber hinaus soll nachfolgend die Notwendigkeit eines Austauschs von Profildaten diskutiert werden, welche später Einfluss auf die Wahl der technischen Mittel zur Datenhaltung nimmt.

### 2.2.1 Grundlagen

Eine zentrale Herausforderung an ein Referenzmodell für Profiling stellt die Frage nach den geeigneten, für alle denkbaren Anwendungsfälle passenden Profildaten dar. Hierzu bedarf es zunächst folgender Definition:

Ein **Profilschema** ist eine Spezifikation, die festlegt, welche individuellen Daten eines Nutzers einer Website in welcher Granularität gespeichert werden und welche gegebenenfalls notwendige logische Struktur zwischen diesen Daten besteht. Dies erfolgt durch Angabe benannter Attribute, die mit Werten definierter Wertebereiche belegt werden können oder die auf andere Attribute verweisen. Aus einem solchen Profilschema lassen sich konkrete Profilinzenzen oder kurz **Profile** ableiten, welche die in Kapitel 1.1.4 geforderte charakterisierende Eigenschaft besitzen.

Das Verhältnis von Profilschema zu Profil gleicht damit dem einer Klasse zu einem daraus instanziierten Objekt im objektorientierten Programmierparadigma.

Profile müssen persistent gespeichert werden können. Dabei sollte neben einem einfachen Datenzugriff auch ein automatisiertes Löschen von Daten nach fest definierten Zeiträumen möglich sein, um die Datenmenge nicht unnötig durch anonyme Profile ansteigen zu lassen oder Datenschutzbestimmungen Rechnung zu tragen.

### 2.2.2 Erweiterbarkeit, Strukturierung und Werteskalierung

Profilschemata können danach unterschieden werden, ob sie bezüglich der in ihnen vorhandenen Attribute statisch oder erweiterbar sind, ob ihre Attributmenge Strukturinformationen enthält und welche Wertebereiche zur Belegung der Attribute verwendet werden.

#### 2.2.2.1 Statische versus erweiterbare Profilschemata

Statische Profilschemata enthalten eine feste Anzahl vorgegebener und über alle Profilinzenzen gleicher Attribute, die mit einem Wert belegt werden können, während bei erweiterbaren Profilschemata neue Attribute hinzugefügt werden können.

Die Vorteile statischer Profilschemata liegen in der einfacheren Realisierbarkeit, tendenziell geringerem, aber zumindest pro Profilinzenz beschränktem Datenaufkommen und in der leichteren Vergleichbarkeit von Profilen.

Demgegenüber führt die dynamische Erweiterung, also die Hinzunahme neuer Attribut-Werte-Paare zu einzelnen Profilen, mit der Zeit zu einer fortschreitenden Spärlichkeit der Profile im mathematischen Sinne.

Beispielsweise kann bei potenziellen Konsumenten das Maß ihres Interesses an n Sachgebieten erfasst und diese Informationen in Anlehnung an [MCS00] als n-dimensionale Vektoren gespeichert werden, wodurch grundlegende Operationen wie der Vergleich zweier Profile auf Standardvektoroperationen zurückzuführen sind.

Wird den Nutzern erlaubt, selbstständig laufend neue Interessensgebiete zu ihrem Profil hinzuzufügen, so ist die Vergleichsmöglichkeit von Profilen auf größte gemeinsame Teilmengen mit Werten belegter Attribute beschränkt.

Erweiterbare Profilschemata weisen jedoch eine größere Flexibilität und eine wesentlich detailreichere Erfassung von Nutzerdaten auf. Gerade die einfache Anpassungs- und Erweiterungsmöglichkeit von Profilen ist vor dem Hintergrund sich schnell wandelnder Anforderungen und Systeme im Bereich elektronisch unterstützter Geschäftsabwicklung besonders wichtig. So sollten Betreiber in der Lage sein, neue Interaktionsmöglichkeiten auf ihrer Website anzubieten und deren Nutzung zur Profildatengewinnung heranzuziehen, ohne tiefgreifende Änderungen am Profilingssystem durchzuführen. Offensichtlich wird dies am Beispiel eines Online-Fachhändlers, der im Rahmen einer lateralen *Diversifikation* eine neue Warengruppe anbietet, die bei der Erstellung des bisher verwendeten Profilschemas nicht berücksichtigt werden konnte, deren Absatz jedoch zu Informationsanreicherung in den Profilen der Käufer führen soll.

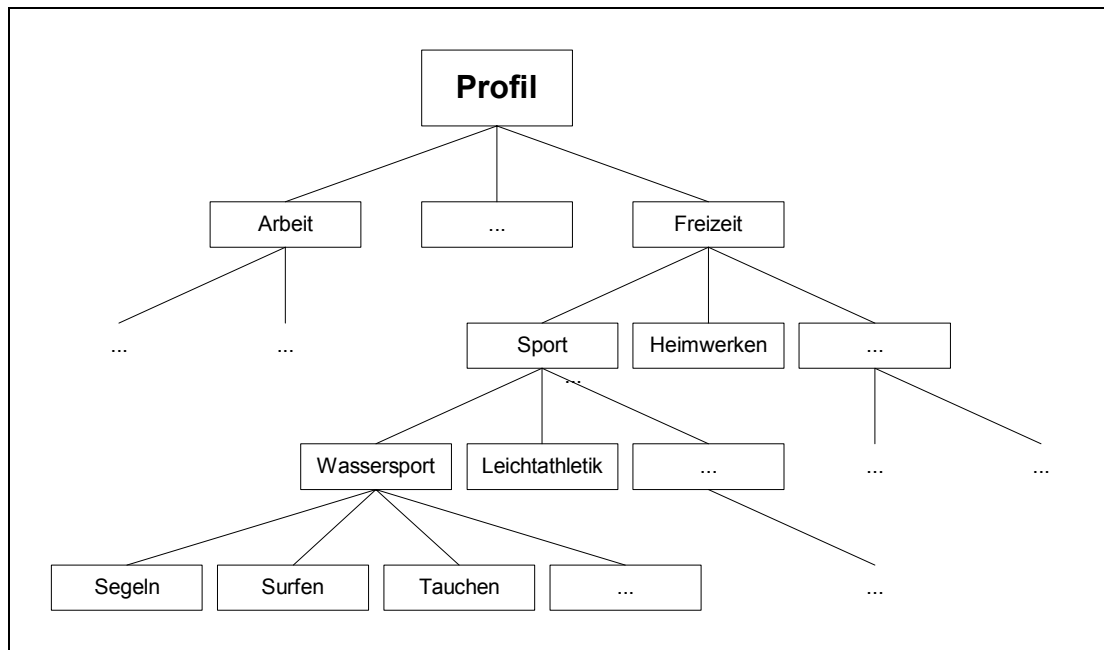
An ein Referenzmodell für Profiling kann damit die Anforderung gestellt werden, sowohl statische als auch erweiterbare Profilschemata zu berücksichtigen. Ersteres stellt keine große Herausforderung dar, da bei einem späteren Systementwurf bzw. dessen Implementierung statische Schemata als einfacher Sonderfall erweiterbarer Schemata gehandhabt werden können. Die Erweiterung eines Schemas sollte darüber hinaus möglichst reibungslos während des Betriebs eines Profilingssystems möglich sein und insbesondere nicht zu Veränderungen oder Verlusten bestehender Daten führen.

### 2.2.2.2 Eindimensionale versus hierarchische Profilschemata

Profilschemata werden eindimensional genannt, wenn die zur Speicherung vorgesehenen Attribute in keinem bestimmten Verhältnis zueinander stehen, d. h. nur aus einer losen Menge bezeichneter Werte bestehen. Begünstigt durch das weit verbreitete relationale Datenbankparadigma und dessen primitiver Anwendung ist dies die häufigste Form der Speicherung von Kunden- und Profildaten; so stehen typische Stammdaten wie Name, Anschrift, Geburtsdatum, Geschlecht usw. als Attribute einer Entität 'Kunde' ohne strukturelle Verbindung nebeneinander und werden i. d. R. über einen eindeutigen Datensatzschlüssel referenziert.

Dagegen bieten hierarchische Profilschemata als Beispiel einer strukturierten Speicherung die Möglichkeit, Informationen in einer Baumstruktur gegliedert abzulegen. Während dies bei den eben erwähnten Stammdaten kaum sinnvoll ist, bietet sich diese Vorgehensweise bei Informationen an, die zu einer Kategorisierung gehören, also einer Mehrdeutigkeit vermeidenden Unterteilung vom Begriffen oder Dingen in möglichst homogene Gruppen und Untergruppen.

Abbildung 2 verdeutlicht dies am Beispiel einer Kategorisierung von Freizeitaktivitäten, wie sie zur Erfassung von Nutzerpräferenzen bei e-Business-Websites sinnvoll eingesetzt werden könnte.



**Abbildung 2: Beispiel einer Kategorisierung**

Eine in dieser Weise strukturierte Speicherung bringt gegenüber einem eindimensionalen Schema einen deutlich höheren Realisierungsaufwand mit, erlaubt jedoch eine detailliertere Erfassung insbesondere von Nutzerpräferenzen, innerhalb derer eine semantische Struktur bzw. vorhandene Ordnungsrelation genutzt werden kann. So ist der Nachteil vieler nicht mit Werten belegter Attributknoten in den unteren Ebenen dadurch kompensabel, dass Belegungen von Unterbäumen im jeweiligen Wurzelknoten quantitativ aggregiert werden. Beispielsweise könnte im Knoten 'Sport' die Summe aller ausgeübten Einzelsportarten hinterlegt werden, um für Vergleichsoperationen auf ein geeignetes Maß für die allgemeine Sportlichkeit des Nutzers zurückgreifen zu können.

Im Hinblick auf die Häufigkeit der Anforderung, Kategorisierungen wie beispielsweise Literaturbereiche, Film- und Musikgenres oder Interessensgebiete verwalten zu können, sollten Systeme zur Profilbildung hierarchische Informationsstrukturen unterstützen.

Technische Möglichkeiten hierfür sowie eine Vorstellung von Standardkategorisierungen finden sich in Kapitel 3.4.

### 2.2.2.3 Nicht-rationale versus rationale Skalen

Ein Profil wird normalerweise gebildet, indem vorgesehenen Attributen Werte zugewiesen werden, die sich aus dem Verhalten eines Nutzers ergeben. In Ausnahmefällen, insbesondere bei frei erweiterbaren Profilschemata, kann auch die alleinige Speicherung eines Wertes eine Profilerweiterung darstellen, ohne dass dieser Wert explizit benannt wurde. So stellen protokollierte URLs einer Session sicherlich Profildaten dar, aus welchen

relevante Informationen meist jedoch nur gewonnen werden können, wenn weitere Maßnahmen zu deren Auswertung ergriffen werden.

Die Bedeutung der in einem Profilschema zu sammelnden Informationen wird häufig durch eine sprechende Benennung der zu speichernden Attribute ausgedrückt, was zu typischen Feldbezeichnungen wie 'Nachname', 'Geburtsdatum' oder 'ist\_Hausbesitzer' führt.

Dies erleichtert bei regelbasierten Personalisierungssystemen die erforderliche Verarbeitung der Daten; eine Anweisung zur Einblendung eines Werbebanners könnte beispielsweise wie folgt aussehen:

```
IF (current_user.ist_Hausbesitzer AND (current_user.Einkommen > 3000)) {  
    show(Banner(Feuerversicherung))  
}  
ELSE {show(Banner(Privathaftpflicht))}
```

Von besonderem Interesse ist die Wahl der Datentypen, aus deren Wertebereichen einzelne Attribute belegt werden können. Ohne auf programmiersprachliche oder standardisierte Datentypen einzugehen, soll neben Zeichenketten zwischen nicht-rationalen und rationalen Skalen unterschieden werden. Nicht-rationale Skalen erfassen Nominal-, Ordinal- und Intervallwerte, erlauben also beispielsweise zwei- oder mehrklassige Unterscheidungen (männlich/weiblich), nicht-metrische Rangordnungen ('Segeln ist interessanter als Surfen und Surfen interessanter als Tauchen') und metrische Rangordnungen (Intelligenzquotient). Rationale Skalen besitzen über die Eigenschaften von metrischen Rangordnungen hinaus einen absoluten Nullpunkt, der es ermöglicht, neben den Abständen zwischen Werten auch mit Verhältnissen rechnen zu können [Meff98]. Diese Eigenschaft ist aus Marketing-Sicht insbesondere bei der Erfassung von Präferenzdaten erforderlich, da die fortlaufende Nutzung einer Website und der darauf befindlichen Inhalte und Angebote zur besseren Auswertbarkeit in absoluten Zahlen quantifizierbar sein sollte.

Deshalb muss ein Referenzmodell für Profiling die Erstellung und Verwendung erweiterbarer und hierarchischer Profilschemata mit rationalen Skalen berücksichtigen. Darüber hinaus soll es eine beliebige Kombination der aufgezählten Möglichkeiten hinsichtlich Erweiterbarkeit, Strukturierung und Werteskalierung erlauben, um für jeden konkreten Anwendungsfall den Entwurf eines optimalen Profilschemas zu ermöglichen.

### 2.2.3 Möglichkeiten des Datenaustauschs

Neben der Möglichkeit des organisationsinternen Datenaustauschs, wie er in Kapitel 2.1.4 beschrieben wird, sollte ein Profilingssystem Schnittstellen zu externen Applikationen bereitstellen, die ebenfalls Profildaten enthalten. Dabei kann es sich um gleichartige Systeme anderer Anwendungsdomänen handeln oder um Fremdsysteme, die andersartige, oft proprietäre Datenformate verwenden.

Im erstgenannten Fall muss ein auf einem Referenzmodell basierendes Profilingssystem eine möglichst einfache Schnittstelle für den Austausch von Profildaten zwischen Instanzen dieses Modells vorweisen. So könnten beispielsweise die Betreiber zweier Websites durch Abgleich ihrer Daten die Profile derjenigen Nutzer anreichern, die bei beiden Sites identifizierbar registriert sind, sofern die Semantik der jeweils zusätzlichen Daten zwischen den verwendeten Profilschemata abbildbar ist oder sich das Schema des Daten importierenden Systems geeignet erweitern lässt.

Für den Profildatenaustausch mit andersartigen Systemen sollte ein Referenzmodell für Profiling Schnittstellen vorsehen, über welche Profilinformatoren aus Standardformaten importiert und eigene Daten in den gleichen Standardformaten exportiert werden können. Einige etablierte Formate zum Austausch von Kundenprofilen werden in Kapitel 3.3 vorgestellt.

Der Datenaustausch mit Profilingssystemen, die keinen verbreiteten Standard unterstützen, erfordert einen entsprechend hohen Aufwand. Dabei muss versucht werden, eine Abbildungsmöglichkeit zwischen dem proprietären Datenformat und einem Standardformat zu finden, für welches Schnittstellen vorhanden sind, oder eine individuelle, direkte Zuordnung der Fremdinformationen an das eigene Profilschema zu verwenden, wobei ein Austausch auf Datenbankebene oder mittels XML-Dateien denkbar ist.

### 2.3 Anforderungen an die Profildatenauswertung

In diesem Abschnitt werden verschiedene Aspekte der Auswertung von Profildaten betrachtet. Dazu gehören Überlegungen zum geeigneten Zeitpunkt möglicher Auswertungen, die Definition sinnvoller Operationen auf einzelnen Profilen sowie eine Vorstellung aus der Marktforschung bekannter Analyseverfahren für Mengen von Profilen.

#### 2.3.1 Profilnutzung und -auswertung zur Laufzeit

Im Hinblick auf die Verwendung von Profildaten zur Personalisierung von Inhalten ergibt sich die Anforderung, während der Benutzung einer Website nicht nur ein Profil für den aktiven Nutzer aufzubauen oder zu erweitern, sondern auch einen lesenden Zugriff auf die bereits vorhandenen Profildaten zu ermöglichen und die stattfindenden Interaktionen zumindest teilweise zur Laufzeit auszuwerten. Dies gilt insbesondere für die notwendige Zuordnung einer Semantik zu abgerufenen URLs, welche erforderlich ist, um beispielsweise Regeln auszuwerten, die bereits zur Personalisierung einer soeben angeforderten Seite genutzt werden sollen.

#### 2.3.2 Operationen auf Profilen

In Anlehnung an [Best00] erscheint es vor allem für Abfragen der Profildatenbank hilfreich, Operationen auf Profilen durchführen zu können. Sei  $P$  die Menge aller möglichen Profile eines gegebenen Profilschemas,  $A$  die Menge aller Attribute dieses Schemas,  $X$  eine rationale Wertemenge, wie sie für die Belegung von numerischen Attributen verwendet werden kann. Denkbar sind nach [Best00] insbesondere folgende Operationen:

- **Merge:**  $(P \times P \rightarrow P)$

Vereinigung zweier Profile zu einem zusammengesetzten Profil. Fortgesetzte Anwendung dieser Operation bzw. eine Realisierung, die eine Menge von Profilen als Eingabe verwendet, führt zu einem Gruppenprofil.

Anwendungsbeispiel: Bestimmte Daten der Einzelprofile einer Gruppe von Nutzern (Mitarbeiter der Einkaufsabteilung eines Unternehmens, Mitglieder einer Familie) können auf diese Weise aggregiert werden, sodass ein Vergleich zwischen den von diesen Gruppen durchgeführten Online-Käufen oder Seitenabrufen möglich wird.

- **Average:**  $(P \times P \rightarrow P)$

Durchschnittsbildung zweier Profile. Analog zum Einsatz von 'Merge' kann mit dieser Operation ein Durchschnittsprofil einer beliebigen Teilmenge der Nutzer errechnet werden.

Anwendungsbeispiel: Zu einer ausgewählten Teilmenge von Nutzern lassen sich so leicht signifikante Kennzahlen wie das Durchschnittsalter, das durchschnittliche Einkommen oder die durchschnittliche Online-Zeit auf den Seiten des Anbieters berechnen.

- **Filter:**  $(P \times A \rightarrow P)$

Filtern von Attributen aus einem Profil. Diese Operation kann verwendet werden, um Profile an ein kleineres Profilschema anzupassen oder besser vergleichbar zu machen.



Anwendungsbeispiel: Die triviale 'Filter'-Funktion dient lediglich der Auswahl von Attributen eines Profils analog zu Angabe von Spaltennamen im `SELECT`-Befehl einer SQL-Anweisung.

- **Weight:**  $(A \times X \rightarrow A)$   
Vergabe einer Gewichtung für Attribute. Verschiedene Gewichtungen für Attribute können benutzt werden, um die Bedeutung einzelner Profile hinsichtlich bestimmter Kriterien zu ermitteln.

Anwendungsbeispiel: Mittels 'Weight' kann innerhalb einer Abfrage der Profildatenbank das Einkommen von Nutzern höher gewichtet werden, um potenzielle Käufer für ein besonders teures Produkt zu ermitteln.

- **Difference:**  $(P \times P \rightarrow X)$   
Berechnen des Unterschieds zwischen zwei Profilen. Diese Operation liefert ein mathematisch exaktes Maß für den Abstand zwischen Profilen.

Anwendungsbeispiel: Hiermit können nach [MCS00] sogenannte 'nearest neighbours' eines Nutzers ermittelt werden, um Empfehlungen auszusprechen, oder die charakterisierende Abweichung eines Profils vom Durchschnittsprofil festgestellt werden.

- **Score:**  $(P \rightarrow X)$   
Bewertung eines Profils hinsichtlich eines Index. Eine derartige Operation kann zur Auswahl besonders wichtiger Profile verwendet werden, sofern eine geeignete Abbildung der Profilinformatoren auf skalare Werte angegeben werden kann.

Anwendungsbeispiel: Der Betreiber einer kommerziell ausgerichteten Website könnte mittels 'Score' die 'Top-1000' der Nutzer ermitteln, um diese mit speziellen Marketingmaßnahmen anzusprechen oder einen Treuebonus auszuschütten. Dazu könnte sich 'Score' der Funktionen 'Filter' und 'Weight' bedienen, um zunächst relevante Attribute von Profilen zu selektieren (z. B. Umsatz, Seitenabrufe, Besuchshäufigkeit) und deren Werte anschließend gemäß gewünschter Zielsetzung zu gewichten.

Eine Umsetzung derartiger Operationen in einem Profilingssystem setzt voraus, dass für die einzelnen Attribute eines gegebenen Profilschemas spezifiziert werden kann, ob und wie die jeweilige Operation auf die Werte wirken soll.

Während 'Merge' und 'Average' bei numerischen Wertebereichen leicht mittels Addition und mathematischer Durchschnittsbildung realisiert werden können, impliziert 'Merge' bei komplexeren Daten wie z. B. Clickstreams die Verkettung von Datensätzen und ist auf Textattributen wie 'Nachname' nicht sinnvoll anwendbar.

Abschließend sei angemerkt, dass die Anwendung der Operation 'Merge' auf die Clickstream-Daten einer Teilmenge oder der Gesamtheit aller Profile in Verbindung mit einfachen Abzählfunktionen alle typischen Auswertungen von Logfiles eines Webserver abdeckt, wie sie in Kapitel 2.1.3.2.1 skizziert werden.

### 2.3.3 Analyse von Profilen

Neben den verhältnismäßig einfachen Operationen auf und Auswertungen von Profilen, die im vorangegangenen Abschnitt vorgeschlagen werden, und die überwiegend auf einzelne

oder kleine Mengen von Profilen angewendet werden können, gibt es eine Reihe von Analysemethoden, deren Einsatz einen Erkenntnisgewinn für die Gesamtheit aller Profildaten verspricht, weil grundlegende Abhängigkeiten oder Zusammenhänge aufgezeigt werden.

Dazu zählen die in der Marktforschung häufig verwendeten multivariaten Analysemethoden, deren Struktur findende oder prüfende Eigenschaften eine Verwendung bei komplexen Profildaten nahe legen. Nachfolgend werden in Anlehnung an [Back00] einige Verfahren genannt und damit untersuchbare Fragestellungen beispielhaft aufgeführt:

- Faktorenanalyse:

Die Faktorenanalyse verdichtet zu untersuchende Variablenmengen (Attribute) auf eine kleinere Anzahl aussagekräftigerer Variablen, sogenannter Supervariablen oder Faktoren, indem Abhängigkeiten zwischen den Einzelvariablen erkannt und zusammengefasst werden.

Anwendungsbeispiel: Zahlreiche Attribute eines Nutzerprofils könnten durch eine geringe Anzahl von Faktoren beschrieben werden, um eine regelbasierte Personalisierung zu vereinfachen. So besteht häufig ein Zusammenhang zwischen Bildungsstand, Beruf und Einkommen einer Person, was durch einen Faktor 'gesellschaftliche Schicht' beschrieben werden könnte. Als weiterer Vorteil ergibt sich, dass solche Faktoren zur Bewertung von Profilen auch greifen, wenn die Werte einzelner Attribute, z. B. Einkommen, aus der Reihe fallen oder nicht verfügbar sind. Darüber hinaus kann angenommen werden, dass bei einer Online-Befragung Nutzer eher dazu bereit und fähig sind, einzelne Aspekte ihrer Person wahrheitsgemäß zu beschreiben, als Einstufungen hinsichtlich komplex aggregierter Faktoren vorzunehmen, beispielsweise die Zuordnung zu einer sozialen Schicht oder zu Lifestyle-Stereotypen.

- Clusteranalyse:

Die Clusteranalyse dient zur Identifikation möglichst verschiedener, aber in sich homogener Gruppen innerhalb einer Gesamtmenge anhand der die einzelnen Elemente unterscheidenden Attribute.

Anwendungsbeispiel: Angewendet auf die Profildaten aller Nutzer eines Portals könnte sich ergeben, dass zwei oder mehr deutlich erkennbare Hauptgruppen von Surfern existieren, deren Verhalten auf verschiedene Schwerpunktinteressen, z. B. Unterhaltung, Information oder Kommunikation, schließen lässt.

- Regressionsanalyse:

Im Gegensatz zu den bisher erwähnten Analysemethoden unterscheidet die Regressionsanalyse wie alle nachfolgend genannten Verfahren zwischen zwei Gruppen von Variablen. Es wird stets die Abhängigkeit eines Wertes von einer Menge unabhängiger Variablen unterstellt. Dabei dient die Regressionsanalyse dem Ziel der Überprüfung des vermuteten Zusammenhangs.

Anwendungsbeispiel: Es kann geprüft werden, welchen Einfluss das Einkommen eines Kunden auf den Wert seiner Online-Bestellungen hat oder ob das häufigere Einblenden des gleichen Werbebanners zu einer Veränderung der *Click-Through-Rate* führt.

- Diskriminanzanalyse:

Während bei der Regressionsanalyse beide Variablenmengen metrisch skaliert sind, prüft die Diskriminanzanalyse den Einfluss metrischer unabhängiger Variablen (z. B. Pageviews, Alter) auf nominal skalierte abhängige Variablen (z. B. Hobbys, Familienstand). Auf diese Weise kann auch die Bedeutung einzelner Attribute auf die Zuordnung von Objekten zu vorgegebenen Gruppen untersucht werden.

Anwendungsbeispiel: Die Diskriminanzanalyse wird häufig im Anschluss an eine Clusteranalyse eingesetzt. Während die Clusteranalyse homogene Teilmengen innerhalb der Nutzer einer Website identifiziert, kann die Diskriminanzanalyse zeigen, welche Attribute in welchem Maße die Einordnung der Nutzer in die gefundenen Cluster erklären. Bezogen auf das o. g. Beispiel einer Separation von Unterhaltung und Information suchenden Surfern könnte sich beispielsweise herausstellen, dass das Attribut Bildungsstand in besonders hohem Maße die Trennung dieser beiden Nutzergruppen bewirkt.

- Varianzanalyse:

Die Varianzanalyse hilft, die Abhängigkeit metrischer Variablen von nominalen Ausgangsdaten zu untersuchen.

Anwendungsbeispiel: Auf diese Weise kann geklärt werden, ob weibliche Internetnutzer signifikant häufiger an Online-Chats teilnehmen als männliche.

Die Integration von Analysemethoden der hier vorgestellten Art gehört wie alle anderen Aspekte der Auswertung von Daten nicht primär zu einem Referenzmodell für Profiling. Als Anforderung muss dennoch berücksichtigt werden, die Durchführung derartiger Analysen prinzipiell vorzusehen und dafür geeignete Profildatenformate anzubieten.

### **2.4 Anforderungen an die Profildatenvisualisierung**

Eine Weiterführung des Grundgedankens, gewonnene Profildaten umfangreich auszuwerten, führt zu der Anforderung, diese Informationen visualisieren zu können. Die Verschiedenheit der in Profilen abgelegten Daten führt dabei zu unterschiedlichen Möglichkeiten der Veranschaulichung. Nachfolgend wird beispielhaft auf Clickstreams, einzelne Profile und Mengen von Profilen eingegangen.

#### 2.4.1 Clickstreams

Eine Visualisierung ist insbesondere für Navigationsmuster sinnvoll, also der Struktur eines Clickstreams, welche sich leicht als Multigraph der besuchten Seiten (Knoten) und der bei der Navigation dazwischen verwendeten Links (Kanten) darstellen lässt. Dabei sollten die (Mehrfach-)Kanten gemäß der zeitlichen Abfolge der benutzten Links nummeriert werden.

Da die grafische Darstellung des Navigationsverhaltens eines einzelnen Nutzers nur bedingt von Interesse ist, wird angeregt, insbesondere die in Kapitel 2.1.3.2.1 erwähnten typischen Navigationspfade einer Vielzahl von Nutzern zu visualisieren. In diesem Fall sollten jedoch keine Mehrfachkanten zwischen Knoten erlaubt werden, stattdessen können Kantenmarkierungen die absolute Anzahl oder eine Prozentangabe der Traversionshäufigkeit eines Links angeben, wie dies in Abbildung 3 beispielhaft dargestellt ist. Eine geeignete Datenstruktur hierfür ist ein Graph, auf den die Operation 'Merge' (s. Kap. 2.3.2) ausgeführt werden kann, indem die Kantenmarkierungen addiert werden.

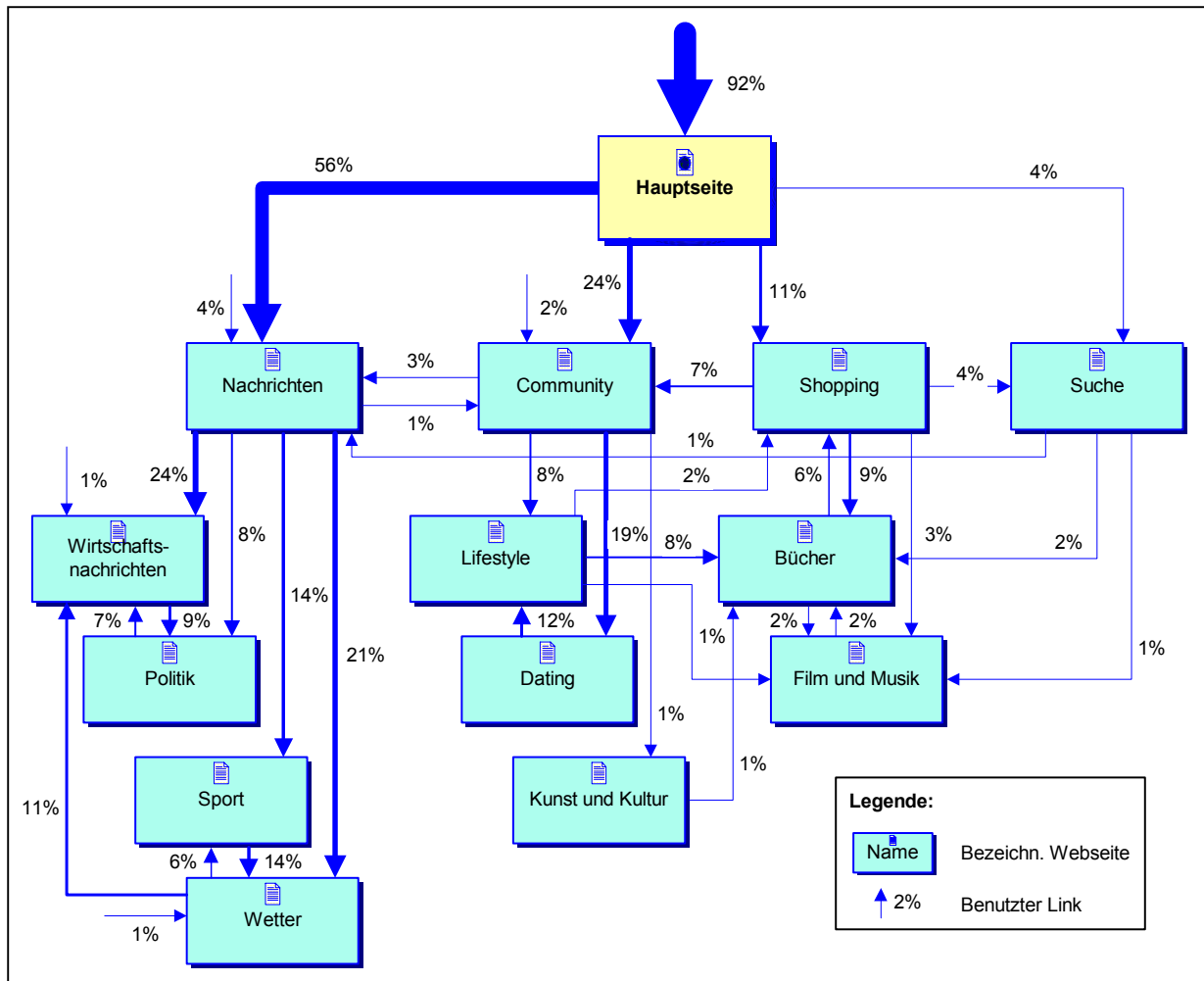


Abbildung 3: Visualisierung von Navigationsverhalten

Abbildung 3 zeigt exemplarisch das Navigationsverhalten auf einer Website, die aus drei Hauptbereichen besteht: einem viergliedrigen Nachrichtenbereich, einem dreigliedrigen Community-Bereich und einem Shopping-Bereich mit zwei Warengruppen. Die Kantenmarkierung gibt an, wie viel Prozent der Nutzer einen Link zwischen zwei Webseiten benutzt haben. Aus Gründen der Übersichtlichkeit sind untergeordnete Seiten und weitere Links nicht dargestellt.

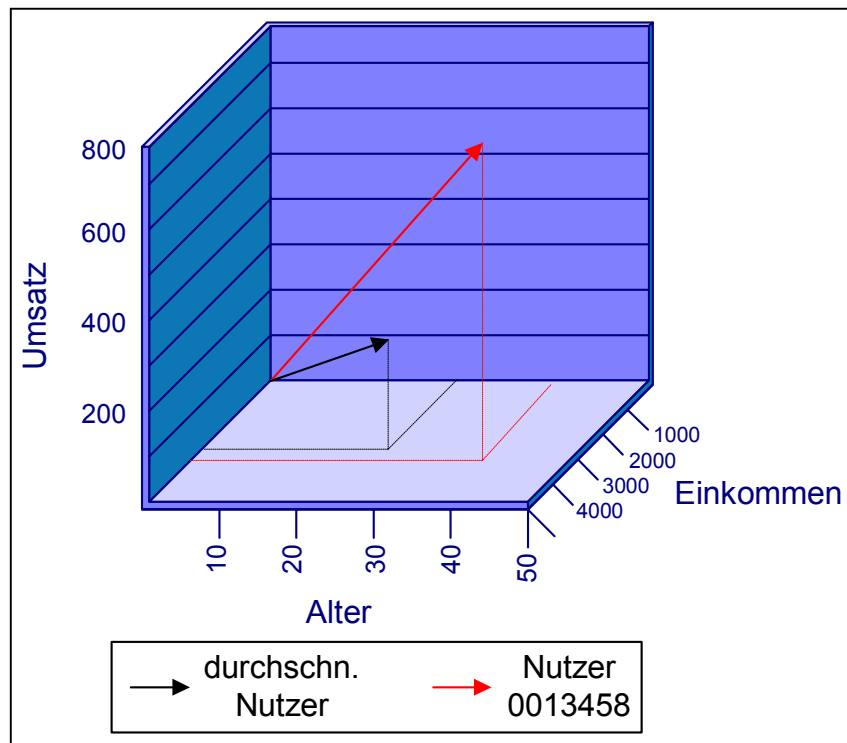
In diesem Fall ist eine deutliche Bevorzugung der Bereiche 'Nachrichten' und 'Community' bei den Nutzern festzustellen, insbesondere 'Wirtschaftsnachrichten', 'Wetter' und 'Dating'. Während zwischen den Bereichen 'Community' und 'Shopping' eine gewisse Affinität innerhalb der Nutzerpräferenzen besteht, interessieren sich nur wenige Nachrichtenleser für die anderen Bereiche.

Die Pfeile ohne Startknoten zeigen an, dass 92% der Nutzer die Website über die Hauptseite betreten, lediglich 2% beginnen ihren Besuch im Community-Bereich und insgesamt 6% benutzen Einstiegspunkte im Nachrichtenbereich.

### 2.4.2 Einzelprofile

Darüber hinaus ist eine Visualisierung der Attributbelegungen einzelner Profile wünschenswert, die über soziodemografische Daten oder Präferenzen Auskunft geben.

Bei einer Beschränkung auf zwei oder drei Attribute mit rationalem Wertebereich lassen sich solche Einzel-, Gruppen- oder Durchschnittsprofile als Vektoren in der Ebene oder im Raum darstellen.



**Abbildung 4: Vektorendarstellung von Profilattributen**

Abbildung 4 zeigt eine Vektorendarstellung von Profilattributen zum Vergleich eines bestimmten Nutzers gegenüber dem Durchschnitt.

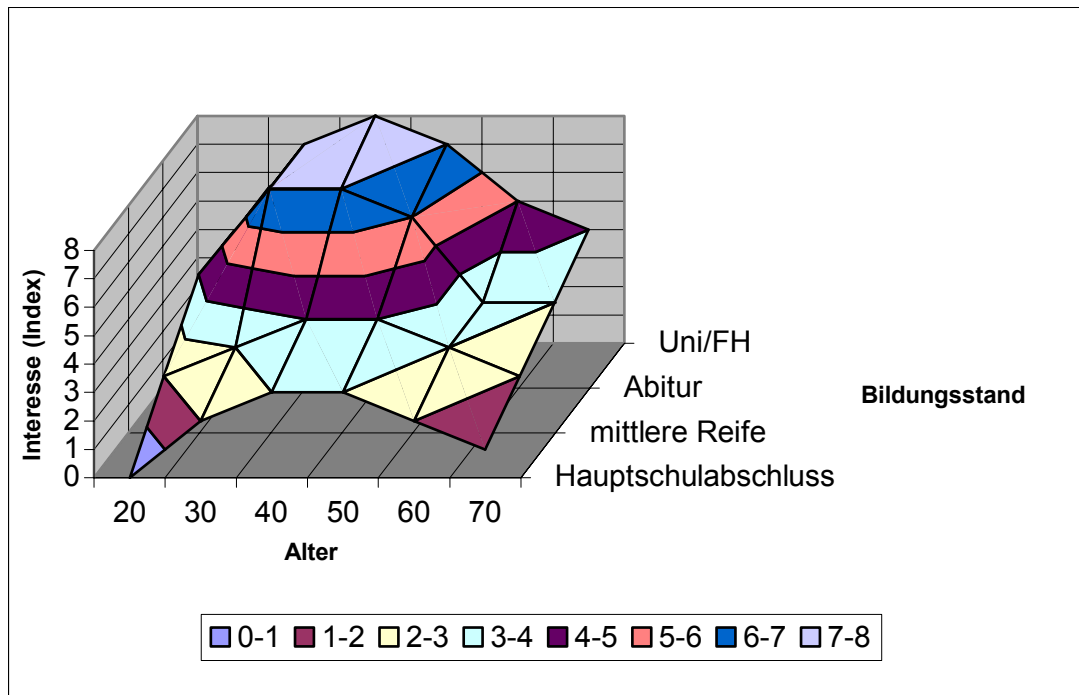
In diesem Beispiel unterscheidet sich der untersuchte Nutzer (ID 0013458) durch ein höheres Alter und insbesondere durch einen deutlich größeren Umsatz auf der Website gegenüber dem Durchschnitt, obwohl sein Einkommen nur geringfügig höher ist.

### 2.4.3 Profilmengen

Schließlich ist eine Visualisierung größerer Profilmengen sowie der Ergebnisse der in Kapitel 2.3.3 vorgeschlagenen Analysen denkbar, sofern sich geeignete Möglichkeiten hierfür finden.

Da die Darstellung von Attributbelegungen als Punkte oder Vektoren in der Fläche oder im Raum bei mehreren Profilen schnell an Übersichtlichkeit verliert, müssen hierfür andere Visualisierungen herangezogen werden. So eignet sich eine Flächendarstellung auch unter Einbeziehung nominaler Werte zur Veranschaulichung aufgedeckter Zusammenhänge.

Abbildung 5 verdeutlicht die hypothetische Abhängigkeit des Interesses von Nutzern an einem bestimmten Themengebiet von deren Alter und Bildungsstand.



**Abbildung 5: Beispiel einer grafischen Darstellung von Abhängigkeiten in Profildaten**

Es wird deutlich, dass das Interesse der Nutzer über alle Altersstufen hinweg größer ist, je höher der Bildungsstand ist. Darüber hinaus ist es unabhängig vom Bildungsstand im mittleren Alter - zwischen 30 und 50 Jahren - am größten, sodass die Kombination dieser beiden Trends gebildete Nutzer dieses Alters als primäre Zielgruppe für den gebotenen Content erscheinen lässt.

### **2.5 Übergeordnete Anforderungen**

Abschließend sollen einige Anforderungen an ein Referenzmodell für Profiling vorgestellt werden, die übergeordnete betriebswirtschaftliche oder technische Aspekte betreffen. Dazu gehören einerseits die leichte Anpassbarkeit an das zugrunde liegende Geschäftsmodell sowie die Möglichkeit einer effektiven Erfolgskontrolle des Profilings und der darauf basierenden Maßnahmen.

Auf der anderen Seite sollte das zu erstellende Referenzmodell derart gestaltet sein, dass eine technische Umsetzung möglichst einfach in vorhandene informationstechnische Infrastrukturen integriert werden kann und darüber hinaus gute Eigenschaften hinsichtlich Performanz und Skalierbarkeit besitzt.

#### 2.5.1 Anpassungsfähigkeit an Geschäftsmodelle

Bei jedem Einsatz von Profiling und darauf basierenden Maßnahmen ist eine sorgfältige Anpassung an bzw. Integration in das zugrunde liegende Geschäftsmodell notwendig. Es gilt, das Profiling den Rahmenbedingungen der elektronisch unterstützen Geschäftsabwicklung auf der Website eines e-Business-Unternehmens anzupassen, als strategische Chance zu nutzen und damit zur Erreichung der unternehmerischen Ziele beizutragen. Aus diesem Grund muss insbesondere ein Modell für Profiling, das den Anspruch eines Referenzcharakters erhebt, ein hohes Maß an Flexibilität bieten, ohne dabei an Effektivität zu verlieren.

Zunächst ist zu prüfen, ob der Einsatz von Profiling im gegebenen Kontext grundsätzlich angebracht ist. Dies ist der Fall, wenn ein erkennbarer Nutzen für den Anbieter zu erwarten ist, wobei sich dieser indirekt aus gesteigertem Nutzen auf Seiten der Nachfrager ergeben kann, beispielsweise in Form verbesserter Bedienbarkeit eines Online-Angebots und daraus abgeleiteter Erhöhung der Kundenzufriedenheit und –bindung. So resümiert [Scho00]: "Properly designed profiling has the potential to help make Web sites more effective and efficient for visitors, while at the same time making them more productive for owners".

Dabei kann es jedoch zu einem Trade-Off von produktivitätssteigernden und kontraproduktiven Effekten kommen, beispielsweise wenn einer wachsenden Anzahl verbesserter Kundenbeziehungen mit höherem *Life-Time-Value* eine Menge nicht erreichter Kunden gegenüber steht, die aus persönlichen Gründen Profiling und Personalisierung kritisch gegenüberstehen oder gar kategorisch ablehnen.

Ein Nutzen kann sich für Anbieter aber auch dadurch ergeben, dass Sammeln und Auswerten individueller Daten als präzises Marktforschungs- oder Controlling-Instrument verwendet werden kann, ohne dass darauf aufbauende Maßnahmen wie beispielsweise Personalisierung ergriffen werden. Profiling bei e-Business-Websites kann insbesondere für die sonst nur schwer durchzuführende Erfolgskontrolle von Marketingaktivitäten dienen, indem beispielsweise die Veränderung des Informationsverhaltens potenzieller Kunden nach umfangreichen (Offline-)Werbemaßnahmen gemessen wird.

Weiterhin muss entschieden werden, für welche Nutzer eines Online-Angebots in welcher Weise Profile aufgebaut werden sollen. Diese Frage ist in Anbetracht der Vielfalt von



Geschäftsmodellen, Website-Konzepten und der zunehmenden Konvergenz derselben von besonderer Bedeutung: das Angebot reicht von stark themenbezogenen Communities über spezialisierte e-Commerce-Angebote, verschiedenste Marktplätze bis hin zu umfassenden Portalen, die mit ihren Channel-, News-, Messaging- und Suchdiensten versuchen, Anlaufstelle für alle Spielarten der Internetnutzung zu sein. In diesem Zusammenhang kann es sinnvoll sein, Profiling nur für eine Untermenge der Nutzer oder einen Teil der Interaktionsmöglichkeiten einzusetzen, wozu hierfür benutzte Systeme in der Lage sein müssen.

In Verbindung mit der Gesamtzielsetzung und den dahin führenden Informationsauswertungen ergibt sich, welche Daten über die Nutzer einer Website grundsätzlich gesammelt werden müssen. Spätestens an dieser Stelle ist einzubeziehen, inwieweit Verknüpfungen zwischen Online- und Offline-Geschäft bestehen und wie diese zu berücksichtigen sind. Die Bedeutung einer ganzheitlichen Sicht auf einzelne Kunden wurde bereits in Kapitel 2.1.4 diskutiert. Ein System für Profiling muss deshalb anpassungsfähig hinsichtlich der zu erhebenden Daten sein und soll Schnittstellen zu anderen Systemen bereitstellen.

Die zu ermöglichende Nutzung und Auswertung der Profilinformatoren erfolgt mit Blick auf die Erreichung der unternehmerischen Ziele und versucht dabei häufig, einen Mehrwert für den Kunden zu schaffen. Grundsätzlich kann zwischen individueller und Massenauswertung der gewonnenen Daten unterschieden werden, wobei eine ausschließliche Massenauswertung wie die Logfile-Analyse keine individuelle Datenerhebung erfordert und deshalb hier nicht weiter betrachtet werden soll.

Darüber hinaus kann unterschieden werden, ob und in welcher Form dem Nutzer einer Website Ergebnisse des Sammelns und Auswertens von Profilinformatoren bekannt gemacht werden. Denkbar sind neben der reinen Auswertung, die keinen direkten Einfluss auf die Kommunikation hat, ein zur Verfügung stellen von Informationen, die aus einer Massenauswertung gewonnen wurden, z. B. Voting-Ergebnisse für redaktionelle Inhalte oder Produktempfehlungen aufgrund des Gesamtkaufverhaltens, sowie eine echte Individualisierung der Kommunikation unter Einbeziehung der persönlichen Nutzerdaten, wie sie von [PeRo96] propagiert wird, beispielsweise mittels Vorselektion von Nachrichtenmeldungen oder Produktempfehlungen anhand bekannter, nutzerspezifischer Präferenzen.

Dabei ist darauf zu achten, dass die bereits erwähnten kontraproduktiven Effekte vermieden werden. So kann die personalisierte Ansprache von Kunden mit ihrem Namen auf der Startseite einer Website dazu führen, dass sich "einige Kunden (...) vom Betreiber zu sehr beobachtet fühlen und einen anderen Anbieter, bei dem sie das Gefühl haben, nicht direkt identifiziert zu werden, vorziehen" [Gart00].

Schließlich ist der Erfolg des eingesetzten Profiling wie in Kapitel 2.5.2 motiviert zu kontrollieren, wobei auch die prinzipielle Einsetzbarkeit und die Güte der Integration des Profiling in das Geschäftsmodell hinterfragt werden sollte. Dabei ist nicht auszuschließen, dass funktionierende Profilingmechanismen als Controlling-Werkzeug eingesetzt werden können, die Schwächen im zugrunde liegenden Geschäftsmodell selbst offenbaren,

beispielsweise indem Kunden ein anderes, der Erreichung der Geschäftsziele entgegenwirkendes Verhalten entwickeln, als bei der strategischen Planung angenommen. Jedem dieser Fälle folgt als Konsequenz, dass Maßnahmen zu ergreifen sind, die den erkannten Missstand auf der entsprechenden Ebene und in angemessener Weise eliminieren, indem also das Profiling, seine Integration oder das Geschäftsmodell selbst modifiziert wird.

Abschließend wird angemerkt, dass trotz der hier stattfindenden Fokussierung auf den Bereich e-Business Profiling und daraus abgeleitete Veränderungen der Kommunikation zwischen zwei Partnern auch im nichtkommerziellen Umfeld grundsätzlich denkbar und in vielen Fällen sinnvoll sind, da beiderseitige Vorteile erreicht werden können. So wie Menschen ein persönliches Gespräch intuitiv an den Kommunikationspartner anpassen und dazu vorhandenes Hintergrundwissen verwenden und erweitern, können automatisiertes Profiling und Personalisierung bei elektronisch unterstützter Kommunikation auch zur Erreichung gesellschaftlicher, sozialer, politischer oder erzieherischer Ziele von Non-Profit-Organisationen wie karitativen Verbänden, Vereinen, Parteien oder öffentlichen Organen beitragen. Beispielhaft sei eine Internet-Community mit der nichtkommerziellen Zielsetzung des Informationsaustauschs ihrer Nutzer genannt, bei der Profiling-basierte Personalisierung die Zielerreichung unterstützen kann, indem die für einzelne Nutzer relevanten Informationen durch Vorauswahl und Empfehlungssysteme besser zugreifbar gemacht werden.

Als Anforderung an ein Referenzmodell für Profiling ergibt sich somit, eine weitreichende Anpassungsfähigkeit an konkrete Geschäftsmodelle zu berücksichtigen, um beim Einsatz davon abgeleiteter Softwaresysteme den oben dargestellten Überlegungen Rechnung tragen zu können.

### 2.5.2 Erfolgskontrolle des Profilings

Der Einsatz von Profilingmaßnahmen und darauf basierender Personalisierung im profitorientierten Umfeld sollte wie alle anderen Geschäftsaktivitäten einer laufenden Kontrolle seiner Effektivität unterzogen werden. Dem zentralen Problem, Metriken und Messverfahren zu finden, die geeignet sind, den Einfluss und Erfolg der angewendeten Maßnahmen zu erfassen und objektiv darzustellen, begegnet [Scho00] bezüglich Personalisierung mit der Aussage: "The metrics required to evaluate success follow directly from the goals. (...) For longer-range goals, such as increasing customer wallet-share and loyalty, measuring the activity of repeat visitors, and shopping cart composition are needed." Ausgehend von den Intentionen des One-to-One- und Relationship-Marketings muss also im Wesentlichen versucht werden, das Maß der Kundenbindung festzustellen, welches durch profilbasierte Personalisierung erreicht wird. Hierzu sollte ein Referenzmodell für Profiling die Ermittlung diesbezüglicher Kennzahlen unterstützen, insbesondere die oben erwähnte Besuchshäufigkeit eines Surfers, seine durchschnittliche Verweildauer und Interaktionsintensität, die Entwicklung der durch ihn erzielten Umsätze und Erträge pro Transaktion und die Summe aller Umsätze und Erträge über die Dauer der Kundenbeziehung.

Zur Quantifizierung des Vorteils der Personalisierung an sich kann ggf. ein Vergleich mit Nutzern gezogen werden, die in ihren persönlichen Privacy-Einstellungen Profiling bzw. Personalisierung abgelehnt haben (vgl. Kapitel 2.1.6).

### 2.5.3 Technische Integrationsmöglichkeit

Als übergeordnete technische Anforderung an ein Referenzmodell für Profiling kann die leichte Integrationsmöglichkeit in bestehende Hardware-/Softwareumgebungen betrachtet werden. Diese ergibt sich aus der Tatsache, dass in den meisten Unternehmen bereits DV-Systeme zur Geschäftsprozessabwicklung eingesetzt werden, welche Kundendaten enthalten, und darüber hinaus auch schon Online-Angebote betrieben werden, die ggf. Content-Management-, Community- oder Shopsysteme integrieren.

In diesen Fällen kann nur selten gefordert werden, aufgrund der Einführung eines Profiling-Systems für Websites eine tief greifende Umstrukturierung oder Neukonzeption einer komplexen, i. Allg. heterogenen DV-Landschaft vorzunehmen, obwohl dies im Hinblick auf eine Zusammenführung aller relevanten Kundeninformationen vielleicht wünschenswert ist. Vielmehr muss angestrebt werden, Profiling möglichst flexibel in solche Umgebungen einzubetten, gewonnene Profildaten anderen Anwendungen verfügbar zu machen, und ebenso von Fremdapplikationen bereitgestellte Daten oder Funktionen zu nutzen.

Schließlich soll vor dem Hintergrund sich rasch ändernder Marktbedingungen und Geschäftsmodelle nicht ausgeschlossen werden, das Profiling-System selbst grundlegend zu erweitern oder gar um zusätzliche Funktionalitäten zu bereichern. Denkbar ist hier insbesondere eine Weiterentwicklung in Richtung einer Personalisierungs- oder CRM-Software, einer Call Center-Applikation oder eines CMS.

### 2.5.4 Performanz und Skalierbarkeit

Die häufig implizit von Serversystemen geforderten Aspekte Performanz und Skalierbarkeit sollen im Bezug auf ein Referenzmodell für Profiling hervorgehoben werden. Ein dieses Modell realisierendes System muss zumindest einen Teil der notwendigen Verarbeitungsschritte auf Profildaten in Echtzeit während der Interaktionen eines Nutzers auf der Website durchführen. Insbesondere eine Personalisierung, die alle zuvor in einer Session erfolgten Clicks berücksichtigt, verursacht Rechenzeitaufwand pro Nutzer und HTTP-Request, und die anschließende dynamische Generierung einer individuell an den Nutzer angepassten HTML-Seite erfordert eine vollständige Interaktions- und Regelauswertung. Beides darf nicht zu einer unzumutbaren Verzögerung bei der Verwendung der Website führen.

Diesem inhärenten Bedarf an Rechenleistung kann einerseits mit einer geeigneten Systemarchitektur und laufzeiteffizienten Implementierung begegnet werden, wofür einige technische Möglichkeiten in Kapitel 3.2 vorgestellt werden. Andererseits müssen für das Profiling bei hochfrequentierten e-Business-Websites die gleichen Skalierungsmöglichkeiten zur Verfügung stehen, welche sich im Bereich der Web- und Applikationsserver-Technologie etabliert haben, um die geforderten Antwortzeiten zumindest mit entsprechendem Aufwand erreichen zu können.

### 3 Technische Grundlagen für Profiling

Nach der Definition verschiedener Anforderungen an ein Referenzmodell für Profiling im vorangegangenen Kapitel erfolgt nun die Vorstellung der als Basis zur Verfügung stehenden Technologien, Protokolle und Standards, wobei auf deren Eignung zur Erfüllung der gestellten Anforderungen eingegangen wird.

Dabei werden zunächst Internettechniken beschrieben, welche Rahmenbedingungen für Möglichkeiten der Profildatengewinnung vorgeben, ehe verschiedene Aspekte der serverseitigen Verarbeitung von Nutzerinteraktionen und Profildaten diskutiert werden. Anschließend erfolgt eine Einführung in die wichtigsten Standards im Kontext von Profiling sowie einige Möglichkeiten zur Bereitstellung von Metainformationen für einzelne Elemente einer Website.

#### 3.1 Internettechniken als Grundlage der Kommunikation

Die für Profiling grundlegende Notwendigkeit der Zuordnung von abgerufenen Inhalten zu einzelnen Internetnutzern erfordert eine kurze Einführung in das verwendete Transportprotokoll und darauf basierenden Möglichkeiten der Identifizierung von Nutzern mittels Cookies und Session-IDs.

##### 3.1.1 Das Hypertext Transfer Protocol

Das Hypertext Transfer Protocol (HTTP) zur Übertragung von HTML-Dateien bildet die technische Kommunikationsgrundlage des WWW, welches als multimedialer Dienst den mit Abstand größten Teil der heutigen Internetnutzung ausmacht und wie die anderen Dienste, insbesondere e-Mail (SMTP, POP3) und FTP, auf dem Internet Protokoll (IP) basiert.

HTTP kann als sehr einfaches Protokoll bezeichnet werden; jeder Anfrage (Request) eines Clients begegnet der Server mit der Rücksendung einer Antwort (Response), welche im Normalfall das abgerufene Dokument, meist eine HTML-Datei, enthält.

Beide Nachrichtentypen bestehen prinzipiell aus drei Blöcken: ein Request enthält zuvorderst einen Befehl (Methode) in Verbindung mit der exakten Bezeichnung der adressierten Ressource, dahinter einen sogenannten Header, der zusätzliche Informationen über den Client, z. B. über akzeptierte Dokumenttypen, enthält und abschließend einen Body, in dem optional Nutzdaten transportiert werden können. Ein Response besteht dagegen aus einem Statuscode, der den Erfolg der Anfrage beschreibt, Header-Informationen und den angeforderten Nutzdaten im Body [Fiel99].

Ein Request verwendet typischerweise die Methode `GET` und bezeichnet die angeforderte Ressource auf dem Server mittels einer Pfadangabe einschließlich Dateiname, wie sie innerhalb einer URL hinter der Host-Bezeichnung und einer optionalen Portnummer vorkommt:

<code>http://</code>	<code>www.p-buy.de</code>	<code>:80</code>	<code>/products/overview.html</code>
Protokoll	Servername oder IP-Nummer	Port (optional)	Ziel-Ressource

Darüber hinaus sieht HTTP die Übermittlung zusätzlicher Daten in Requests vor, wie sie üblicherweise durch das Ausfüllen der Felder eines HTML-Formulars anfallen, und die serverseitig mithilfe spezieller Techniken, z. B. CGI (s. u.), an die benannte Ressource übergeben werden. Diese kann ein beliebiges ausführbares Programm oder eine Datei sein, deren Aufruf zu einer Verarbeitung

der zusätzlichen Daten führt und die den Inhalt des Response erzeugen, in der Regel also HTML-Code, der vom Client-Browser empfangen, interpretiert und dem Nutzer angezeigt wird.

Zur Übertragung dieser Zusatzdaten kann die bereits erwähnte Methode `GET` verwendet werden, wobei die Informationen dann als Attributnamen-Werte-Paare durch das Zeichen '?' getrennt an die Bezeichnung der Ziel-Ressource gehängt werden. Die Bekanntgabe des Vor- und Zunamens über ein Kontaktformular an eine JavaServer Page (JSP) könnte beispielhaft folgendermaßen aussehen:

```
http://www.p-buy.de/info/contact.jsp?f_name=Matthias&l_name=Dorka
```

Die Übertragung größerer Datenmengen erfolgt jedoch bevorzugt unter Verwendung der Request-Methode `POST`, bei der die Attributnamen-Werte-Paare ohne vorangestelltes '?', jedoch getrennt durch '&'-Zeichen den Header-Informationen in einem eigenen Datenblock nachgestellt werden [SpEc99].

Die Einfachheit des HTTP-Protokolls impliziert, dass ein Protokollieren jeglicher Interaktion eines Nutzers auf einer Website prinzipbedingt allein durch das Abfangen und Speichern aller Requests, insbesondere der an `GET` und `POST` gebundenen Informationen, möglich ist.

Es wird auch ersichtlich, dass HTTP ein zustandsloses Protokoll ist, d. h. dass keinerlei Speicherung von Informationen, die einen Kontext der Kommunikation über eine einzelne Request-Response-Folge hinaus definieren könnte, unterstützt wird. Daher ist es notwendig, zur Realisierung komplexerer Abläufe, wie dem Füllen eines virtuellen Warenkorbs mit anschließender Online-Bestellung, eine server- oder clientseitige Datenhaltung zu entwickeln, die eine Speicherung zusätzlicher, den Kommunikationskontext beschreibender Informationen zulässt. Möglichkeiten hierfür werden in den Kapiteln 3.1.2 und 3.2.1 vorgestellt.

Abschließend sollen noch einige Problemfelder diskutiert werden, die sich aus der heutigen Struktur des Internets und verbreiteter technischer Details ergeben.

Moderne Browser verwenden zur Vermeidung unnötiger Wartezeiten und Reduzierung der tatsächlich übertragenen Datenvolumen einen lokalen Zwischenspeicher für Dateien, Cache genannt, der sowohl HTML-Seiten als auch darin eingebettete Elemente wie Bilder enthält, welche bevorzugt gelesen und dem Nutzer angezeigt werden, wenn angenommen werden kann, dass sich diese an ihrem Ursprungsort (URL) in keiner neueren Version befinden. Dies geschieht insbesondere bei der Verwendung der in vielen Browsern vorhandenen Vor- und Zurück-Buttons, die ein wiederholtes Betrachten der zuletzt besuchten Seiten erlauben. In solchen Fällen wird kein erneuter Request an den Server gesendet, sodass dort auf diese Weise nicht eindeutig rekonstruiert werden kann, welche Seiten wie lange tatsächlich zwischen zwei registrierten Requests angeschaut wurden.

Dieses Caching wird darüber hinaus von einigen Proxies angewendet, also Rechnern, die beispielsweise Unternehmensnetze an das Internet anschließen oder von Providern zwischen Endkunden und Backbone geschaltet werden, und die eingehende Requests dahin gehend untersuchen, ob die angeforderte Ressource nicht bereits im Zwischenspeicher liegt. Ist dies der Fall, so wird die lokale Kopie an den Client geschickt und der ursprüngliche Request nicht an den Zielservers weitergeleitet.

Die Möglichkeiten zur Vermeidung beider Caching-Verfahren bestehen einerseits in der Übermittlung bestimmter Header-Informationen im Client-Request und Server-Response, welche Caching-Mechanismen deaktivieren oder die Gültigkeitsdauer von Dokumenten auf

eine sehr kurze Zeitspanne herabsetzen sollen, aber deren Beachtung in den Implementierungen der Browser und Proxies nicht gewährleistet werden kann, andererseits in der Verwendung dynamisch generierter Seiten und Session-IDs (s. Kapitel 3.1.3), die zu stets verschiedenen Zielressourcen im Request sorgen, was Caching-Mechanismen unwirksam macht.

#### 3.1.2 Cookies

Die von Netscape eingeführte und mittlerweile von nahezu allen Browsern unterstützte Technik des Setzens und Lesens von Cookies erlaubt es, kleine Datenmengen auf dem Client persistent zu hinterlegen.

Dies geschieht mittels einer `Set-Cookie`-Anweisung im Response-Header des Servers, der eine Reihe von Attributnamen-Werte-Paaren sowie eine Gültigkeitsdauer und ein Gültigkeitsbereich angehängt werden können. Letzterer bestimmt den URL-Bereich, an den bis zum Ablauf der Gültigkeitsdauer der Cookie-Inhalt automatisch bei jedem Client-Request übertragen wird. Somit ist ein Auslesen des Cookies i. d. R. nur dem Server erlaubt, der es gesetzt hat [SpEc99].

Damit bilden Cookies eine der Möglichkeiten, nutzerspezifische Daten zu speichern, die den Kontext einer Verbindung z. B. mittels einer Session-ID (vgl. 3.1.3) beschreiben können und erlauben unter gegebenen Umständen die Identifizierung eines Nutzers bei einem späteren Website-Besuch. Obwohl diese Technik von vielen Systemen sogar für essentielle Daten wie Warenkorbhalte intensiv genutzt wird, muss davor gewarnt werden, sich ausschließlich auf das Setzen von Cookies zu verlassen. Wichtige Daten zum Navigations- und Transaktionsverhalten können nur auf dem Server sicher und jederzeit verfügbar gehalten werden, da folgende Probleme bei der Verwendung von Cookies bestehen:

- Nutzer können ihren Browser so einstellen, dass keine Cookies gesetzt werden dürfen.
- Ggf. führt der Versuch, Cookies zu setzen, zu Warn- bzw. Bestätigungsmeldungen im Browser, die einen Nutzer verunsichern können.
- Einige Browser unterstützen das automatische Löschen von Cookies nach Beendigung einer Internet-Session; Cookies können von Nutzern auch jederzeit manuell gelöscht werden.
- Je nach verwendeter Browser-Betriebssystem-Kombination kann eine personenindividuelle Unterscheidung von Nutzern nicht angenommen werden. Im ungünstigsten Fall einer Surf-Station in einem Internet-Café kann es vorkommen, dass Hunderte von Menschen innerhalb kurzer Zeit unter einem Browserprofil das Web nutzen, und somit die in Cookies gespeicherten Informationen über zahlreiche Nutzer verschmelzen, was zu Fehlinterpretationen führt.

Im umgekehrten Fall muss davon ausgegangen werden, dass einige Surfer mehr als einen Rechner nutzen, z. B. auf der Arbeit und zu Hause, was zur Vermeidung von in mehrere Cookies aufgespaltenen Datenbeständen impliziert, mit einer Authentifikation per Nutzerkennung und Passwort sowie serverseitiger Datenhaltung zu arbeiten, sofern dies hinsichtlich der Akzeptanz bei den Nutzern durchsetzbar erscheint.

Dennoch sollten Cookies zusätzlich eingesetzt werden, um zumindest zu versuchen, stets wiederkehrende Nutzer einer Website zu identifizieren, über die keine weiteren Erkennungsmerkmale verfügbar sind, da zu vermuten ist, dass die meisten Surfer einen bestimmten Rechner bevorzugt verwenden bzw. viele Rechner nur von einer einzelnen oder wenigen Personen benutzt werden.

### 3.1.3 Session-IDs

Neben der Identifizierung bekannter Nutzer beim erneuten Besuch einer Website ist es nach den Überlegungen aus Kapitel 3.1.1 auch notwendig, die zusammengehörenden Requests eines einzelnen Nutzers während einer Session als solche zu erkennen. "Man bezeichnet das Einordnen von Anfragen und Antworten an einen Web-Server in einen bestimmten Kontext als Sitzungsverfolgung (session tracking) und den Kontext selbst als Sitzung (session) [Balz00]."

Zu diesem Zweck können eindeutig identifizierende Zeichenketten, sogenannte Session-IDs (SID), eingesetzt werden, die meist ein aus Daten wie IP-Adresse, Zeitstempel oder laufender Nummer des Requests abgeleiteter Hash-Wert sind. Weiter reichende Überlegungen zum Einsatz global eindeutiger Kennungen und sicheren Methoden zur Erzeugung von Session-IDs finden sich in [HaCo96], brauchen an dieser Stelle jedoch nicht aufgegriffen zu werden.

Die Verbindung zwischen einzelnen Requests und einer zugeordneten Session-ID lässt sich am einfachsten durch die im vorangegangenen Kapitel beschriebenen Cookies erreichen: mit dem ersten Response des Servers an einen Client wird die neu erzeugte SID in einem Cookie abgelegt und bei jedem nachfolgenden Request an diesen Server zurück übermittelt. Da der Einsatz von Cookies nicht immer möglich ist, existiert zusätzlich die Möglichkeit, die SID in die URLs eines Webangebots zu codieren. Diese als Session-Encoding bezeichnete Maßnahme ist jedoch sehr aufwändig, da sie die dynamische Generierung von Webseiten voraussetzt, bei der zudem jeder in einer Seite vorkommende Link um die SID als GET-Parameter ergänzt werden muss (URL-Rewriting) bzw. jedes HTML-Formular ein zusätzliches Input-Attribut mit der SID erhalten muss, sofern die Folgeseite wieder zur gleichen Session gehören soll.

In den meisten Anwendungsfällen für Profiling wird zwecks Personalisierung jedoch ohnehin mit Systemen zur dynamischen Generierung von Webseiten gearbeitet (vgl. Kapitel 3.2.1), welche in aller Regel den Einsatz von Session-IDs umfangreich unterstützen.

### 3.1.4 Web Bugs

Neben Cookies und Session-IDs bilden sogenannte 'Web Bugs' eine weitere Möglichkeit, das Verhalten von Nutzern auf Websites zu protokollieren. Als Web Bugs werden meist unsichtbare Grafiken bezeichnet, die mittels <IMG>-Tag in HTML-Seiten, HTML-codierte e-Mails oder andere HTML-fähige Dateien wie z. B. Microsoft Word-Dokumente eingebettet werden und beim Öffnen der Datei von einem Internetserver geladen werden.

Ist der Rechner eines Nutzers zu diesem Zeitpunkt mit dem Internet verbunden, wird hierdurch ein HTTP-Request ausgelöst, der alle in Abschnitt 3.1.1 vorgestellten Header-Informationen einschließlich vorhandener Cookies an den Server überträgt. Dies führt zu

einem Hit im Logfile des Webserver, von dem die unsichtbare Grafikdatei, oft auch als 'clear gif' bezeichnet, nachgeladen wird (vgl. Logfile-Analyse in Kapitel 1.2.4).

Nachfolgend ist der HTML-Code zweier Web Bugs dargestellt, wovon der erste den Abruf einer Webseite von [www.quicken.com](http://www.quicken.com) bei einem Server des Werbevermarkter DoubleClick registriert.

```

```

```
<IMG WIDTH=1 HEIGHT=1 border=0  
SRC="http://media.preferences.com/ping?ML_SD=IntuitTE_Intuit_1x1_RunOfSite_A  
ny&db_afcr=4B31-C2FB-10E2C&event=reghome&group=register&time=1999.10.27.20.5 6.37">
```

Neben der IP-Adresse des Clients, der URL der umgebenden Seite und den Header-Informationen können auch weitere Daten mit dem Request versendet werden, wie das zweite Beispiel verdeutlicht. Es werden verschiedene Attribut-Werte-Paare übermittelt, die eine bestimmte Interaktion, eine Datenbankinformation und einen Zeitpunkt codieren.

Auf diese Weise kann die Protokollierung individueller Clickstreams externalisiert werden, indem speziell präparierte Web Bugs eines Dienstleisters in die HTML-Seiten einer Website integriert werden. Der besondere Vorteil von Web Bugs liegt zum einen in der Möglichkeit, Sammelprofile mehrerer Websites zu erstellen, wenn der Fremdserver eines einzelnen Dienstleisters verwendet wird, zum anderen besteht die Chance, das Öffnen von e-Mails und Dateien zu überwachen. Dies ermöglicht eine Erfolgskontrolle beim Versand von Werbe-e-Mails, bei der es unter Verwendung von Cookies zudem möglich ist, die e-Mail-Adresse eines Nutzers einem evtl. vorhandenen Profil zuzuordnen.

Web Bugs und das dahinter stehende Prinzip werden zz. von Vermarktern von Online-Werbeflächen wie DoubleClick ([www.doubleclick.com](http://www.doubleclick.com)) intensiv eingesetzt, um Werbebanner effektiver einzublenden, indem beispielsweise häufige Wiederholungen des gleichen Banner-Motivs vermieden werden und zudem der Besuch verschiedener Websites durch denselben Nutzer protokolliert wird, sofern deren Betreiber Kunde bei DoubleClick sind.



### **3.2 Servertechniken für die Profildatenverarbeitung und –speicherung**

Neben der Client-Server-Kommunikation im Internet spielen die technischen Möglichkeiten der Verarbeitung von Client-Anfragen und damit einher gehender Maßnahmen wie Profildatensammlung oder Personalisierung auf Seiten des Servers<sup>4</sup> eine wichtige Rolle im Hinblick auf ein Referenzmodell für Profiling. Deshalb werden in diesem Abschnitt hierfür vorhandene Techniken, Paradigmen und Vorgehensweisen diskutiert, die auch Aspekte der Ressourcenschonung, Lastverteilung und Datenhaltung betreffen, um den in Kapitel 2 aufgestellten Anforderungen gerecht zu werden.

#### **3.2.1 Serverseitige Dynamik**

Die Weiterentwicklung von Webservern als einer speziellen Form von Fileservern hin zu dynamischer Aufbereitung und Verarbeitung von Daten kann als fortschreitender Prozess betrachtet werden, bei dem mit jeder neuen Entwicklungsstufe einerseits die Schwächen der vorhergegangenen Technik eliminiert wurden und andererseits durch Vereinigung verschiedener Funktionalitäten in einer Software die Grenzen klassischer Multi-Tier-Architekturen enger zusammenrückten.

##### **3.2.1.1 Common Gateway Interface**

Die Einführung des *Common Gateway Interface* (CGI) bot in Verbindung mit HTML-Formularen die erste Möglichkeit, Webseiten dynamisch zu erzeugen und dabei Eingaben eines Nutzers zu verarbeiten. Hierzu werden die über die eigentliche URL hinausgehenden Eingabedaten eines Requests (vgl. Kapitel 3.1.1) vom Webserver als Standardeingabe an ein beliebiges ausführbares Programm weitergeleitet, das als Standardausgabe Header-Informationen und HTML-Code produziert, welche an den Client zurück gesendet wird. Insbesondere die Skriptsprachen Perl und PHP haben sich für diesen Zweck etabliert und sind heute noch weit verbreitet.

Neben dem prinzipiellen Vorteil der individuellen Anpassung der Seite an den Aufruf, z. B. Anzeige ausgewählter Informationen aus einer Datenbank, besteht die Möglichkeit, per Umgebungsvariablen auf die Header-Informationen des Requests zuzugreifen, welche bereits für Profiling nutzbare Informationen enthalten können.

Es ist jedoch negativ zu bewerten, dass CGI wenig performant ist, da jeder Aufruf einen eigenen Betriebssystemprozess auf dem Server erzeugt und die inhärente Zustandslosigkeit des Protokolls HTTP nicht ohne weiteres überwunden werden kann [OHE99].

##### **3.2.1.2 Webserver-Erweiterungen**

Die Bemühungen die o. g. Nachteile des Common Gateway Interface zu eliminieren, münden in der Vermeidung des Abspaltens neuer Betriebssystemprozesse je Client-Anfrage. Während 'Fast CGI' das Wiederverwenden eines Betriebssystemprozesses ohne dessen Terminierung erlaubt, werden vor allem Erweiterungen der Webserver eingesetzt, die es

---

<sup>4</sup> Als serverseitig wird hier die gesamte technische Infrastruktur bezeichnet, die typischen Clients (Browsern) auf der anderen Seite des Kommunikationsmediums Internet gegenüber steht (Webserver, Applikationsserver, Datenbankserver), auch wenn es sich hierbei um verteilte Umgebungen handelt, innerhalb derer wiederum eine Aufteilung in Clients und Server vorgenommen werden könnte.

ermöglichen, mehrere Requests innerhalb desselben oder eines anderen Serverprozesses zu verarbeiten. So kann beispielsweise die Fähigkeit zur Verarbeitung von Perl-Skripten oder PHP-Seiten in den weit verbreiteten Webserver Apache per Modul eingebunden (Unix/Linux) oder einkompiliert werden, während Produkte von Microsoft und Netscape durch Nutzung der von diesen bereitgestellten APIs (ISAPI, NSAPI) erweitert werden können. Diese Möglichkeit wird auch von zahlreichen anderen Herstellern genutzt, um proprietäre Techniken wie Cold Fusion von Allaire zu vertreiben.

In der gleichen Weise können Microsoft Active Server Pages (ASP) und JavaServer Pages (JSP, von JavaSoft spezifiziert), verwendet werden; zwei Techniken, die das Einbetten von Quellcode in HTML-Dateien erlauben, welcher beim Abruf der Seiten ausgeführt wird und damit die dynamische Variation von Inhalten ermöglicht. Beide bilden de-facto-Standards, wobei zumindest JSP herstellerübergreifend unterstützt wird und welche im Unterschied zu den zuvor genannten Techniken jeweils ein etabliertes Komponentenmodell zur Softwareentwicklung im Hintergrund haben, zu dem sie kompatibel sind. Active Server Pages benutzen als Standardsprache Visual Basic Script und können mit COM-Objekten interagieren, die i. d. R. in Visual Basic implementiert sind, während JavaServer Pages standardmäßig Java verwenden und damit grundsätzlich interoperabel mit JavaBeans und Enterprise JavaBeans (EJB) sind. Somit können diese Techniken einer komplexen komponentenbasierten Applikation als äußerste serverseitige Präsentationsschicht zur Generierung von HTML für den Client dienen, ohne einen Technologiebruch zu erfordern [Balz00]. Weiterführende Informationen zur konkreten Realisierung einer solchen Architektur in der JSP/Java/EJB-Welt finden sich beispielsweise in [FiKo00], [Kass00], [Roma99].

Durch die Verarbeitung der Requests in einem Prozess und Speicherbereich ergibt sich neben der deutlich besseren Performanz der Vorteil, dass ein Interaktionskontext leicht verwaltet werden kann, ohne einen umständlichen und langsamen Umweg über das Speichern und Lesen von Informationen in Datenbanken oder im Dateisystem zu machen. Die meisten der oben genannten Verfahren unterstützen dies direkt, indem sie automatisch ein Session-Objekt anlegen und verwalten, welches einen einzelnen aktiven Nutzer identifizierbar repräsentiert und eine Referenzierung weiterer zusätzlicher Datenobjekte zur Laufzeit erlaubt. Dies kommt den Anforderungen an Performanz eines Profilingystems und darauf basierender Personalisierung entgegen.

#### 3.2.1.3 Applikationsserver

Mit der Zunahme der serverseitigen Verarbeitungsmöglichkeiten von Daten für eine Webanwendung wird ein kontinuierlicher Übergang zu Applikationsserver-Technologien vollzogen. Während ASP, JSP und Cold Fusion ursprünglich als Webserver-Erweiterung betrieben wurden, existiert mittlerweile auch Serversoftware, die speziell auf die Abarbeitung dieser Kleinstprogramme ausgelegt ist und dabei die primitive Funktionalität eines Webserver integriert. Schließlich entstanden Applikationsserver verschiedener Hersteller und unterschiedlichen technologischen Ursprungs, die z. T. konkrete Komponentenmodelle unterstützen und beispielsweise wie zahlreiche EJB-Container die Verarbeitung von HTTP-Requests mit Servlets und JavaServer Pages implizit beherrschen [OHE99].

Darüber hinaus sind Applikationsserver zum Betrieb komplexer komponentenbasierter Anwendungen geeignet, indem sie die Laufzeitumgebung und Basisfunktionalitäten für Komponenten bereitstellen, die den Vorgaben eines bestimmten Komponentenmodells folgen, sowie ggf. Schnittstellen zu anderen Komponentenmodellen anbieten.

Somit können grundsätzliche Vorteile der komponentenbasierten Softwareentwicklung wie Plattformunabhängigkeit und ein hohes Maß der Wiederverwendung erreicht werden, wenn Komponenten die nach [GrTh00] unter anderem angestrebten Eigenschaften des wohldefinierten Zwecks, der Kontextunabhängigkeit, Portabilität sowie Integrations- und Kompositionsfähigkeit besitzen.

Die in Kapitel 2 aufgestellten Anforderungen an ein Referenzmodell für Profiling bei e-Business-Websites und die in diesem Kapitel beschriebenen technischen Grundlagen dafür legen ein komponentenorientiertes Vorgehen bei der Umsetzung des zu erstellenden Modells in einen Softwarefeinentwurf und dessen anschließende Implementierung nahe.

Der Einsatz eines Applikationsservers kann dabei das für Profiling erforderliche Session-Management als auch die Erfüllung weiterer Anforderungen erleichtern, beispielsweise hinsichtlich Performanz, Skalierbarkeit und Persistenz.

#### 3.2.2 Realzeit- und Vorabauswertungen

Die Anforderungen einer Profilnutzung und –auswertung zur Laufzeit aus Kapitel 2.3.1 und die teilweise komplexen Operationen und Analysen auf Profilen aus den Kapiteln 2.3.2 und 2.3.3 lassen es sinnvoll erscheinen, zwischen Realzeit- und Vorabauswertungen zu unterscheiden.

Während der lesende und schreibende Zugriff auf Profilinformatoren sowie eine einfache Semantikauswertung des Clickstreams einer Session zwecks Personalisierung zur Laufzeit möglich sein müssen, ist kaum realisierbar, rechenintensive Analysen auf der Gesamtheit oder einer großen Teilmenge der Profile ad hoc auszuführen.

Aus diesem Grund wird vorgeschlagen, komplexe Auswertungen wie Faktoren- oder Clusteranalysen zeitgesteuert in Phasen geringer Aktivität auf der Website durchzuführen, beispielsweise nachts. Dennoch darf der Ablauf einer Analyse eine Fortführung des Profiling nicht grundsätzlich blockieren, sodass Mechanismen für einen potenziell gleichzeitigen Lese- oder Schreibzugriff auf Profilinformatoren durch einen Analyse- und einen Trackingprozess vorzusehen sind. In aller Regel werden derart rechenintensive Auswertungen, die einen stark wissenskomprimierenden Charakter haben, ohnehin nur in größeren zeitlichen Abständen durchgeführt.

Da der Erkenntnisgewinn einer solchen Analyse jedoch dem Profilingssystem selbst wieder verfügbar gemacht werden soll, wird auch angeregt, einerseits zusätzliche 'statistische' Profile anzulegen und zu verwalten, andererseits nutzerspezifische Auswertungsergebnisse als eigene Attribute eines realen Profils vorzuhalten.

Auf diese Weise kann ein mittels 'Merge' (vgl. Kapitel 2.3.2) aus allen realen Profilen zusammengestelltes Profil die absolute Summe aller Seitenabrufe und den Gesamtumsatz aller Transaktionen der Website repräsentieren, während ein entsprechendes 'Average'-Profil Aufschluss über das Durchschnittsalter und –einkommen der Nutzer gibt, worauf dann beispielsweise zur
--

Auswertung einer Personalisierungsregel mit den Standardmethoden zum Auslesen von Attributwerten eines Profils zugegriffen werden kann.

Besonders interessant erscheint diese Überlegung in Verbindung mit einem Gruppenkonzept, also der Möglichkeit, die Zugehörigkeit eines Profils zu einer oder mehrerer Gruppen zu speichern sowie auch der Gruppe Attribute zuzuweisen und mit Werten zu belegen. Hiermit kann die Zugehörigkeit eines Einzelnen zu einer mithilfe der Clusteranalyse ermittelten homogenen Teilmenge aller Nutzer abgebildet werden, deren Durchschnittsprofil dieses Cluster im Sinne eines Stereotypen hervorragend repräsentiert.

Die Speicherung von Referenzen auf ausgewählte Summen- oder Durchschnittsprofile sowie vorab berechnete 'nearest neighbors' eines realen Einzelprofils kann dann leicht für Empfehlungsmechanismen benutzt werden.

#### 3.2.3 Lastverteilung

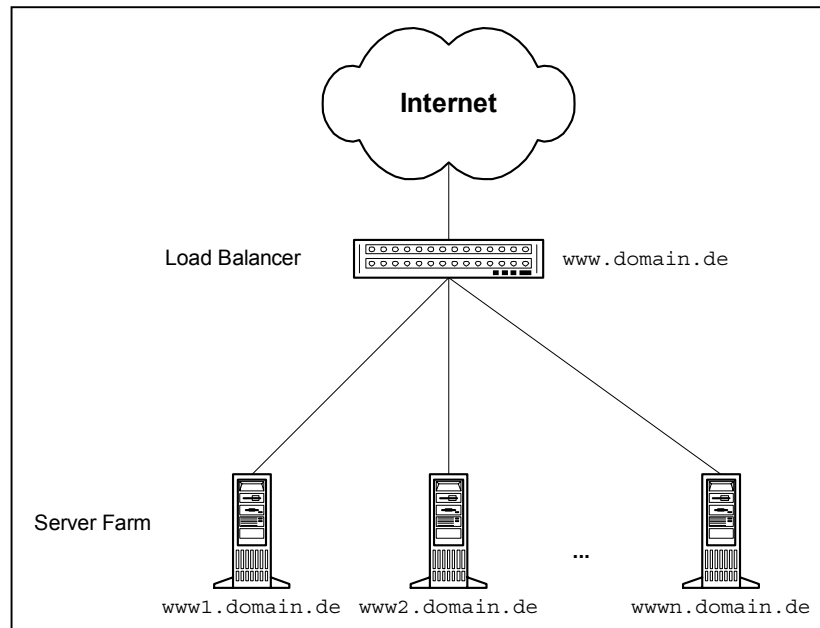
Den in Kapitel 2.5.4 aufgestellten und für die meisten Softwaresysteme genannten Anforderungen an Performanz und Skalierbarkeit kommt im Bereich der Webapplikationen eine besondere Bedeutung zu. Es gilt, den Nutzer durch kurze Antwortzeiten des Systems und raschen Seitenaufbau involviert zu halten; unzumutbare aufgezwungene Verlangsamungen bei der Ausübung eines Web-basierten Workflows führen zu einer rapide sinkenden Akzeptanz des Dienstes bei den Nutzern.

Dieser Problemstellung kann nicht in allen Belangen begegnet werden, so hängt die effektive Antwortzeit u. a. von der physischen Distanz zwischen Client und Server ab und eine hohe Netzlast in Teilen des Internets verringert die Datenübertragungsrate, obwohl der Server performant agiert.

Um zu vermeiden, dass unzureichende Antwortzeiten durch Überlastung des serverseitigen Systems aus Web- und/oder Applikationsserver sowie Profilingmechanismen verursacht werden, kommt nach der technischen Aufrüstung und Optimierung eines einzelnen Servers vor allem die Verteilung der anfallenden Last auf mehrere Server in Betracht.

Im Fall einfacher Webserver ist es ausreichend, eingehende HTTP-Requests auf gleichartige Systeme mit gespiegelten Datenbeständen zu verteilen (vgl. Abbildung 6).

Diese Verteilung kann reihum geschehen (round robin), sollte sich jedoch bevorzugt an der Auslastung der einzelnen Server orientieren und damit eine echte Lastverteilung (load balancing) realisieren. Am Markt verfügbare Lösungen erlauben die Konfiguration der Verteilung z. B. anhand der Systemlast, Antwortzeit oder Verbindungsanzahl der einzelnen Server.



**Abbildung 6: Lastverteilung zwischen Webservern**

Darüber hinaus kann eine Verteilung gemäß der angeforderten URL des Requests vorgenommen werden, um beispielsweise prozessorzeitintensive CGI-Aufrufe an hierfür optimierte Maschinen zu delegieren, während FTP-Downloads von speziellen Fileservern gehandhabt werden.

Beim Ablauf von Webapplikationen, bei denen ein Verbindungskontext erhalten werden muss, ist es notwendig, zusammengehörende Requests desselben Clients demselben Server zuzuordnen. Dies kann durch den Load Balancer u. a. dadurch sichergestellt werden, indem nach Zuweisung eines Servers dessen Kommunikation direkt mit dem Client erfolgt. Der Zugriff auf zugrunde liegende Geschäftsdaten erfolgt bei derartig verteilten Webapplikationen entweder durch Verwendung eines gemeinsamen Datenbankservers oder ebenfalls verteilte Server mit geteilten oder gespiegelten Daten.

Ein weiterer Vorteil von Systemen zur Lastverteilung ist, dass sie bei entsprechender Auslegung auch zur Erhöhung der Ausfallsicherheit beitragen können.

#### **3.3 Standards im Kontext von Profiling**

Nachfolgend werden einige Standards erläutert, die im Zusammenhang mit Profiling von Bedeutung sind, da sie sich entweder mit einzelnen Aspekten dessen befassen oder ein allgemeines Hilfsmittel hierfür darstellen.

So widmet sich die *Platform for Privacy Preferences* dem Schutz der Privatsphäre von Konsumenten bei der Nutzung von Websites, während die *Electronic Commerce Modeling Language* eine Vereinfachung von Online-Käufen anstrebt. *Customer Profile Exchange* und *Open Profiling Standard* zielen auf den vereinheitlichten Austausch von Profilinformatoren ab.

Viele dieser Standards verwenden die universelle Beschreibungssprache *Extensible Markup Language* (XML), welche auch für den Austausch von Profildaten benutzt werden kann und sich darüber hinaus für die in Kapitel 3.4 beschriebenen Aufgaben eignet.

##### 3.3.1 Platform for Privacy Preferences

Die Spezifikation 'Platform for Privacy Preferences 1.0' (P3P1.0) ist am 15.12.2000 vom W3C in den Status 'Candidate Recommendation' erhoben worden und stellt damit einen wichtigen Vorschlag für einen ersten, sich möglicherweise etablierenden Standard für die Wahrung der Privatsphäre von Internetnutzern dar.

"The Platform for Privacy Preferences Project (P3P) enables Web sites to express their privacy practices in a standard format that can be retrieved automatically and interpreted easily by user agents" [Cran00].

Damit ist P3P kein Standard für Profiling, berührt es jedoch einerseits dadurch, dass profilbildende Maßnahmen eines Website-Betreibers im Falle der Berücksichtigung von P3P in einer entsprechenden Privacy Policy ausgedrückt werden müssen, andererseits der Vorschlag u. a. die Speicherung und den Austausch von erweiterbaren Nutzerdaten vorsieht, welche diesen beschreiben. Dazu verwendet P3P XML sowohl für die Beschreibung der in jedem Fall erforderlichen Privacy Policy einer Website als auch für die sogenannten 'data schemas', welche Struktur und Inhalt ggf. clientseitig gespeicherter Nutzerinformationen und -präferenzen definieren.

Die Anwendung von P3P kann folgendermaßen skizziert werden: Internetnutzer benötigen einen 'User Agent', beispielsweise einen per Plug-In P3P-fähigen Browser, in dem sie ihre persönlichen Präferenzen hinsichtlich Schutz der Privatsphäre einstellen. Dies beinhaltet vor allem, welche Informationen in welchem Maße und zu welchem Zweck der Nutzer zu offenbaren bereit ist, angefangen von anonymen Daten, wie sie ohnehin im Logfile des Webserver anfallen, über Kontakt- und Zahlungsinformationen bis hin zu individuellen und zuordenbaren Analysen des Nutzerverhaltens und die Weitergabe solcherlei Informationen an Dritte.

Websitebetreiber haben dagegen die Art der Gewinnung von Daten sowie deren geplante Verwendung in standardkonformen Privacy Policies zu beschreiben, wobei verschiedene Bereiche der Website bis hin zu einzelnen Seiten mit verschiedenen Policies verknüpft sein dürfen. So kann für einen Shopbereich die Bekanntgabe von Name und Postanschrift des Nutzers gefordert werden, während reine Informationsseiten der gleichen Website nur per

Logfile-Analyse zur Verbesserung der Benutzbarkeit untersucht werden. Zur Referenzierung der jeweils zutreffenden Privacy Policy sieht P3P drei Möglichkeiten vor:

1. Die Privacy Policy befindet sich an einer 'well-known location', wofür der Pfad `/w3c/p3p.xml` eines jeden Webhosts vorgeschlagen wird.
2. Die Privacy Policy zu einer bestimmten Webseite wird per `link`-Tag in den HTML-Code eingebunden.
3. Jeder Response-Header (s. Kapitel 3.1.1) des Servers beinhaltet eine spezielle P3P-Header-Information, in der die URL der Privacy Policy angegeben ist, die zur im Request angeforderten Ressource gehört.

Schließlich liegt es an der Implementierung der P3P-Clientsoftware, was geschehen soll, wenn die Privacy Policy einer angeforderten URL die Maßgaben der Privacy Preferences eines Nutzers überschreitet. Denkbar ist hier das Anzeigen einer Warnmeldung im Browserfenster, welche die Art der Überschreitung umgangssprachlich beschreibt und dem Nutzer die Auswahl lässt, die Ressource trotzdem anzufordern oder den Request abzuberechnen.

Zur Beschreibung von Struktur und Inhalt der Informationen, auf die sich eine Privacy Policy beziehen kann, verwendet P3P frei definier- und erweiterbare Datenschemata, wobei ein konkretes Basisschema (base schema) vorgegeben wird, das folgende Elemente enthält:

- `user`: beschreibt den Nutzer u. a. mit Namen, Geburtsdatum, Geschlecht sowie privaten und geschäftlichen Kontaktinformationen.
- `thirdparty`: beschreibt beteiligte Dritte mit den gleichen Attributen wie bei `user`, über die auf diese Weise Informationen ausgetauscht werden können.
- `business`: kennzeichnet eine Organisation durch deren genaue Bezeichnung und Kontaktinformationen. `business` wird insbesondere zur Beschreibung des Websitebetreibers und damit des Ausstellers von Privacy Policies verwendet.
- `dynamic`: beschreibt eine Reihe zusätzlicher Informationen, welche überwiegend im Kontext der Session entstehen, z. B. `clickstream`, `clientevents`, `cookies`, `searchtext` und weitere.

Dabei basieren einige Elemente der o. g. Basisschemata auf vorgegebenen Basisdatenstrukturen (basic data structures), welche im Sinne der Spezifikationen möglichst häufig zur Bildung neuer und Erweiterung existierender Datenschemata verwendet werden sollen.

Beispielsweise verwenden die Kontaktinformationen von `user` und `business` die Datenstruktur `contact`, welche sich wiederum aus `postal`, `telecom` und `online` zusammensetzt, um die Postanschrift, verschiedene Telefon- und Telefaxnummern sowie e-Mail-Adresse und URL eines Beteiligten zu beschreiben [Cran00].

Interessant im Hinblick auf Profiling ist, dass P3P vorschlägt, häufig nachgefragte unveränderliche Daten, wie sie beispielsweise das Basisschema `user` beinhaltet, optional clientseitig zu speichern und eine zumindest halbautomatische Übertragung an den Server zu unterstützen. Dies kann nach [Cran00] beispielsweise dadurch geschehen, dass

Websitebetreiber Inputfelder in HTML-Formularen gemäß P3P-Basis schemata in Punktnotation benennen, also `user.bdate.ymd.year` für ein Feld, in das ein Nutzer sein Geburtsjahr eingeben soll, wobei ihn ein P3P-fähiger Browser mittels automatischer Ausfüllfunktion unterstützen könnte.

Darüber hinaus existiert mit [CLM01] ein Arbeitsentwurf für eine 'P3P Preference Exchange Language' (APPEL), die u. a. einen weiter automatisierten Austausch von Präferenzdaten zwischen Client und Server zum Ziel hat, um auf den Nutzer besser zugeschnittene Ausgaben zu erzeugen, die dessen Präferenzen entsprechen.

Zur Erstellung eines Referenzmodells für Profiling sollte deshalb beachtet werden, mindestens solche Attribute für Nutzerprofile vorzusehen, welche die von P3P vorgeschlagenen Basis schemata ausnutzen oder überdecken. Zur Erreichung einer höheren Nutzerakzeptanz könnte P3P auch grundsätzlich berücksichtigt werden, beispielsweise indem sich einstellbare Stufen der Profiling-Intensität an P3P orientieren und die automatische Generierung von Privacy Policies integriert wird.

#### 3.3.2 Electronic Commerce Modeling Language

Die Electronic Commerce Modeling Language (ECML) ist ein offener Standard zur Spezifikation von Formularfeldern auf Bestellseiten von e-Commerce-Websites. Die Verwendung einheitlicher Namen und Längen der Eingabefelder für Name, Anschrift und Kontaktinformationen des Bestellers, der Rechnungs- und Lieferanschrift sowie der Kreditkartendaten soll es so genannten 'digital wallets' ermöglichen, einmal erfasste Eingaben eines Nutzers in Bestellformularen automatisch einzusetzen, um Online-Bestellungen einfacher und bequemer zu machen [ECML01a].

Zur Wahrung weitest gehender Kompatibilität mit P3P werden dabei Datenfelder und Feldbezeichner in Anlehnung an die Basis schemata und Basisdatenstrukturen aus [Cran00] gewählt. [Coco99] regelt darüber hinaus, wie ECML in Verbindung mit P3P benutzt werden kann, indem für in ECML zusätzlich vorkommende Datenfelder, z. B. Kreditkarteninformationen, P3P-konforme XML-Schemata angegeben werden.

Obwohl ECML trotz Beteiligung namhafter Unternehmen wie IBM, Microsoft und Sun nicht die ursprünglich angestrebte Verbreitung im Bereich des e-Commerce gefunden zu haben scheint und eine Weiterentwicklung zur Spezifikationsversion 1.1 bis heute ausblieb (Stand 03/2001), unterstreicht es durch die Integration mit P3P dessen Bedeutung insbesondere für standardbildende Organisationen und liefert ein eingängiges Anwendungsbeispiel für P3P-konforme XML-Datenschemata.

#### 3.3.3 Customer Profile Exchange

Da seitens der im Bereich e-Business tätigen Unternehmen neben der Notwendigkeit von Profiling auch der Nutzen eines anbieterübergreifenden Datenaustauschs von Kundenprofilen erkannt wurde, bilden sich erste Standards, die diese Aufgabe herstellerunabhängig lösen wollen.

Besonders erwähnenswert ist Customer Profile Exchange (CPEXchange), eine Vereinigung von mehr als 70 Firmen, die unter Führung von IBM einen gleichnamigen Standard etablieren möchte. Ende Oktober 2000 wurde mit [BoHo00] die Version 1.0 einer



Spezifikation veröffentlicht, an der u. a. Allaire, ATG, IPlanet, Net Perception und Vignette mitgewirkt haben. Sie sieht insbesondere eine erweiterbares Datenmodell für Kundenprofile vor, welches verschiedenen Anforderungen an Datenschutz und Wahrung der Privatsphäre, insbesondere P3P, gerecht werden soll. Der Datenaustausch basiert auf XML und soll unabhängig von konkreten Transportmechanismen sein.

Die Spezifikation von CPExchange beinhaltet ein komplexes Datenmodell zur Abbildung von an Geschäftsprozessen Beteiligten. Dabei kann es sich sowohl um natürliche Personen als auch um Organisationen handeln, welche sich nicht unbedingt in einer Anbieter-Endkunde-Beziehung befinden müssen, sodass die Bezeichnung 'Kundenprofil' (Customer Profile) als sprachliche Vereinfachung gesehen werden kann.

Wesentliche Bestandteile des Datenmodells aus [BoHo00] sind:

- Privacy Information  
Diese enthalten die Elemente `PURPOSE`, `RETENTION` und `ACCESS` aus P3P, welche den Verwendungszweck, die Vorhaltdauer und die Zugriffsmöglichkeit eines Nutzers auf die eigenen Daten beschreiben.  
Für verschiedene Teile eines Nutzerprofils können verschiedene Privacy-Aussagen gemacht werden.
- Descriptive Information  
Hier wird beschrieben, ob das Profil zu einer Einzelperson oder zu einer Organisation gehört, und in welchem Verhältnis eine Person ggf. zu einer Organisation oder Gruppe steht. Darüber hinaus werden Kontaktinformationen gespeichert, wobei im Hinblick auf das Customer Relationship Management multiple Kontaktpunkte zwischen Beteiligten unterstützt werden, sowie demografische Daten und Angaben über Aktivitäten eines Profilinhabers.
- Interaction History Information  
Die Protokollierung aller Interaktionen zwischen Beteiligten geschieht mit den Objekten `ActionEvent`, `Encounter` und `CaseFolder`. Ein `ActionEvent` beschreibt feingranular eine einzelne Aktion, beispielsweise einen Pageview. Eine Reihe zusammen gehörender Aktionen werden als `Encounter` aggregiert, wobei dies neben Web Sessions auch Anrufe in Call Centern oder persönliche Gespräche einschließt. Schließlich können Aktionen auch so genannte `CaseFolder` zugeordnet werden, die beispielsweise einzelne Anfragen, Angebote oder Aufträge eines Kunden repräsentieren.
- Preference Information  
Diese beschreiben Präferenzen eines Beteiligten, wobei zur Speicherung von Attributnamen und –werten frei definierbare Zeichenketten vorgesehen sind. Es kann vermerkt werden, ob eine Präferenz auf einer Selbstbeschreibung des Beteiligten, Angaben Dritter, Beobachtungen oder eigenen Schlussfolgerungen basiert.

Damit wird CPExchange einer Reihe von Anforderungen an zu speichernde Daten innerhalb eines Profils gerecht, insbesondere ist das Bestreben ersichtlich, dem selbst gesetzten Anspruch zu genügen: "This standard builds common ground with emerging web-based

Internet Relationship Management (IRM) and Enterprise Relationship Management (ERM) approaches to online business, decision support, and personalisation" [BoHo00].

Das Maß der Anerkennung und Verbreitung dieses Standards unter e-Business-Unternehmen wird die strategische Entscheidung beeinflussen, inwieweit die Vorgaben dieser Spezifikation für eine Realisierung von Profilingmaßnahmen zu berücksichtigen sind.

Für ein Referenzmodell für Profiling kann das vorgeschlagene Datenmodell als Ausgangspunkt eines universelleren Datenmodells dienen, um zur Beteiligung am unternehmensübergreifenden Austausch von Kundenprofilen die von CPExchange spezifizierten Attribute möglichst einfach in das Austauschformat XML exportieren und auch aus diesem importieren zu können. Sofern ein eigenes Datenmodell nicht eine Obermenge der für einen Datenaustausch minimal erforderlichen Attribute bildet, bedarf es geeigneter Abbildungen zwischen den Informationsschemata.

In nahezu allen Fällen ist die Ausnutzung der von CPExchange vorgesehenen Erweiterungsmöglichkeiten notwendig, da einerseits spezifische Profilattribute einer bestimmten Anwendungsdomäne benötigt werden und sich andererseits Struktur und vorgegebene Belegungen einiger Standardattribute für den europäischen Kulturraum zu sehr an amerikanischen Verhältnissen orientieren, was beispielsweise an der Menge der Hobbysportarten ersichtlich wird.

#### 3.3.4 Open Profiling Standard

Ein ähnlicher Vorstoß zur Etablierung eines Standardprofils durch FireFly Networks, Netscape und Verisign führte bereits im Juni 1997 zur Vorlage eines Grobkonzepts mit der Bezeichnung Open Profiling Standard (OPS) beim *World Wide Web Consortium* (W3C), welches sich jedoch nicht über den Status 'Note' hinaus entwickelt hat.

OPS sieht eine kleine Anzahl fester Attribute vor, die typische Stammdaten wie Name, Adresse, Telefonnummer, e-Mail-Adresse u. ä. eines Nutzers enthalten, sowie eine hierarchische Sammlung weiterer Daten, die als so genannte 'Sections' und 'Subsections' eines Profils bezeichnet werden. Diese Bereiche sollen serverseitig beschrieben werden können, wobei ein späteres Auslesen nur dem jeweiligen Nutzer sowie denjenigen Beteiligten möglich ist, denen es der Daten Schreibende erlaubt. Dieses Prinzip schließt die Stammdaten ein, deren Eigentümer der Profilinhaber ist, und der damit als einziger das Recht besitzt, diese Daten zu ändern [Hens97].

Trotz der Ergänzung der OPS-Framework-Beschreibung um Vorschläge zur Implementierung und Anwendung des Standards fand bisher keinerlei Weiterentwicklung der Dokumente statt. Da die grundlegenden Ideen von OPS zudem von P3P und CPExchange überdeckt und maßgeblich erweitert werden, kann an dieser Stelle auf eine weitere Betrachtung verzichtet werden.

### **3.4 Metainformationen für Elemente einer Website**

Eine zentrale Herausforderung an ein Referenzmodell für Profiling stellt die Klassifizierung<sup>5</sup> bereitgestellten Contents hinsichtlich seiner Semantik dar, um das Verhalten einzelner Nutzer nicht nur beobachten sondern auch rechnergestützt interpretieren zu können. Nachfolgend werden deshalb einige Grundlagen der Beschreibung von Bedeutung mittels Metainformationen für Elemente einer Website eingeführt und konkrete Ideen für deren Umsetzung bei statischen und dynamisch erzeugten HTML-Seiten sowie Transaktionen vorgestellt.

#### **3.4.1 Grundlagen**

In diesem Abschnitt wird dargelegt, dass Nutzerprofile zur Verwendung bei e-Business Websites über die Haltung von Stammdaten und die einfache Protokollierung von Seitenabrufen hinausgehen müssen. Dazu werden einige Möglichkeiten und Standards vorgestellt.

##### **3.4.1.1 Motivation**

Die systematische Zuweisung einer Bedeutung zu einzelnen Elementen einer Website ist entscheidend für das Erstellen charakterisierender Profile. Erlaubt die Analyse der Webserver-Logfiles zwar die Ermittlung häufig besuchter Seiten und typischer Navigationsmuster, so ist eine schnelle Interpretation der Daten für Menschen nur bei statischen HTML-Seiten mit aussagekräftigen Pfad- und Dateinamen möglich (Bsp.: [www.kauf-mich.com/shop/shoes/overview.html](http://www.kauf-mich.com/shop/shoes/overview.html)). Der zunehmende Einsatz von CMS und dynamisch generiertem HTML bewirkt jedoch, dass die in Logfiles vorzufindenden Daten nur noch schwer oder unzureichend ausgewertet werden können, da die verwendeten URLs automatisch generiert werden bzw. kryptische Bezeichner enthalten.

Darüber hinaus ist die Frage zu beantworten, in welcher Granularität welche Nutzungsdaten gesammelt werden sollen, was sich aus der Zielsetzung und der Anwendungsdomäne des Profiling ergibt. So kann es für ein Nachrichten-Portal ausreichend sein, die groben Interessensgebiete seiner Nutzer als feste Einordnung in eine zehn Punkte umfassende Kategorisierung zu speichern, während ein Online-Händler für Angelzubehör möglicherweise einige hundert Präferenzen hinsichtlich Ausrüstung, Fanggewohnheiten usw. verwalten möchte. Der Differenzierungsgrad der zur Einstufung von Content verwendeten semantischen Schemas ist also auf die Granularität der Nutzerprofile abzustimmen. Eine feingliedrige Klassifikation einzelner Ressourcen muss zumindest auf ein weniger komplexes Profilschema abbildbar sein, wenngleich eine eindeutige eins-zu-eins-Beziehung zwischen diesen am nützlichsten erscheint.

Eine umgangssprachliche Beschreibung der Bedeutung einer Web-Ressource (HTML-Seite, Grafik, Audio- oder Videodatei) ist nicht sinnvoll, da diese praktisch kaum rechnergestützt verarbeitet werden kann und die Redundanz natürlicher Sprache zu semantischer Unschärfe

---

<sup>5</sup> Der Begriff 'Klassifizierung' bezeichnet nachfolgend die Tätigkeit des Einordnens von Elementen in vordefinierte Kategorien, deren Gesamtheit 'Klassifikation' genannt wird.

führt. Deshalb existieren Klassifikations- und Zuordnungssysteme, wie sie in Kapitel 3.4.1.2 näher erläutert werden, welche eine Kategorisierung bedeutungsvoller Entitäten in disjunkte Mengen erlauben. Grundsätzlich sollte bei der Verwendung eines solchen Systems die Zuordnung mehrerer Bedeutungen zu einer Entität vorgesehen werden, da Dinge des täglichen Lebens selten nur einer einzelnen semantischen Kategorie zugeteilt werden können, so impliziert beispielsweise das Recherchieren eines Website-Nutzers nach einem Autoradio sowohl das Vorhandensein eines Kfz als auch Interesse an Musik.

#### 3.4.1.2 Dublin Core

Bei der Konzeption eines Personalisierungssystems in [CDA00] wird *Dublin Core* (DC) für die semantische Beschreibung des Inhalts einer Webressource vorgeschlagen. Dieser von der *Dublin Core Metadata Initiative* (DCMI) entwickelte offene Standard ermöglicht die einheitliche und maschinenlesbare Beschreibung einer Internet-Ressource mittels bestimmter Zusatzinformationen, welche beispielsweise Titel, Urheber, Thema und Beschreibung, aber auch Informationen über das Datenformat, Urheberrechte und einen Identifizierungsschlüssel enthalten und als *Dublin Core Metadata Element Set* (DCMES) bezeichnet werden [DCMI01a].

Tabelle 2 zeigt die Elemente des DCMES:

<b>DCMES-Elementname:</b>	<b>Verwendungszweck:</b>
Title	Name einer Ressource
Creator	Urheber der Ressource (Person, Organisation oder Dienst)
Subject	Thema der Ressource (Schlüsselwörter oder Klassifikation)
Description	Beschreibung der Ressource (Zusammenfassung, Inhaltsverzeichnis o. ä.)
Publisher	Veröffentlichende Instanz der Ressource (Person, Organisation oder Dienst; nicht zwangsläufig mit dem Urheber identisch)
Contributor	Personen, Organisationen oder Dienste, die zum Inhalt der Ressource beigetragen haben
Date	Bedeutsames Datum im Lebenszyklus der Ressource
Type	Typus einer Ressource, die ihren Charakter beschreibt (DC schlägt hierfür eine feste Menge von Typbezeichnungen vor, z. B. 'Collection', 'Event', 'Image', 'Software', 'Sound' oder 'Text')
Format	Physikalisches oder digitales Format einer Ressource (z. B. MIME-Type und Größe bei computerlesbaren Ressourcen)
Identifizier	Eindeutiger Schlüssel zur Identifizierung einer Ressource, welcher jedoch kontextabhängig sein kann (z. B. URI für Internetressourcen oder ISBN für Bücher)
Source	Bezeichnung einer Ressource, aus der die zu beschreibende Ressource abgeleitet wurde
Language	Sprache des Inhalts einer Ressource
Relation	Beziehung zu einer anderen Ressource (z. B. inhaltliche Bezugnahme, Abhängigkeit, Zugehörigkeit)
Coverage	Angabe des Geltungsbereichs einer Ressource, insbesondere räumlich oder zeitlich (z. B. Geltungsdauer oder -gebiet)
Rights	Angabe von Rechten bezüglich der Ressource (z. B. Copyright)

**Tabelle 2: DCMES-Elemente für Metainformationen – Version 1.1 [DCMI01b]**

Jedes dieser Elemente wird wiederum selbst durch eine genau festgelegte Reihe von Attributen beschrieben, um die Bedeutung des Elements zu definieren. Dazu zählen beispielsweise Angaben zur Version des DCMES und zur maximalen Anzahl des Vorkommens eines solchen Metainformationen enthaltenden Elements. Mehrfache Verwendung ist bei den vorgesehenen Verfeinerungen der Elemente möglich, so kann 'Date' gemäß [DCMI01c] in den Ausprägungen 'Created', 'Valid', 'Available', 'Issued' und 'Modified' für die Beschreibung verschiedener wichtiger Zeitpunkte im Lebenszyklus einer Ressource genutzt werden.

Darüber hinaus schlägt [DCMI01c] zahlreiche existierende Codierungen für die Belegung verschiedener Elemente vor, für das Element 'Subject', welches im Hinblick auf die thematische Einordnung einzelner Webseiten anhand ihres Inhalts zwecks Ermittlung von Nutzerpräferenzen beim Profiling von besonderem Interesse ist, wird auf folgende Klassifizierungssysteme verwiesen:

- Library of Congress Subject Headings (LCSH)
- Medical Subject Headings (MeSH)
- Dewey Decimal Classification (DDC)
- Library of Congress Classification (LCC)
- Universal Decimal Classification (UDC)

Für die allgemeine Anwendung erscheinen insbesondere DDC, LCC und UDC interessant, wobei es sich um Klassifizierungssysteme handelt, die überwiegend in Bibliotheken zur semantischen Einordnung der Bestände verwendet werden, diesen Zweck potenziell jedoch auch für Webressourcen wie HTML-Seiten und darin eingebettete multimediale Inhalte erfüllen. Die Klassifikation erfolgt i. d. R. durch Vorgabe eines hierarchischen Systems von Nummern, Buchstaben oder Schlagworten, mit denen versucht wird, alle thematischen Aspekte von Veröffentlichungen bzw. Medien sinnvoll strukturiert abzudecken.

Die von Melvin Dewey bereits 1876 erstmals veröffentlichte Dewey Decimal Classification (DDC) arbeitet mit Dezimalziffern und unterteilt in ihrer aktuellen Version DDC 21 auf der obersten Ebene folgende Gebiete (Main Classes):

<b>Nummer:</b>	<b>Themengebiet:</b>
000	Computers, information, & general reference
100	Philosophy & psychology
200	Religion
300	Social sciences
400	Language
500	Science
600	Technology
700	Arts & recreation
800	Literature
900	History & geography

**Tabelle 3: Hauptklassen der Dewey Decimal Classification [DDC01a]**

Die jeweils zweite und dritte Ziffer der verwendeten Nummer wird relevant bei der Unterteilung dieser Klassen in sogenannte Divisions und deren weitere Aufspaltung in Sections. So bezeichnen innerhalb der Klasse 500 (Wissenschaft) die Nummern 510 Mathematik, 520 Astronomie, 530 Physik usw., wobei letztgenannte allgemeine Abhandlungen zur Physik bezeichnet, 531 klassische Mechanik, 532 Flüssigkeitsmechanik, 533 Gasmechanik usw.

Nach der dritten Dezimalstelle kann die Klassifizierung hierarchisch fortgeführt werden, was bewirkt, dass direkte Überbegriffe eines bestimmten Begriffs stets eine um die letzte Stelle

gekürzte Nummer besitzen bzw. diese keine Relevanz hat, gleichrangige und verwandte Begriffe durch eine gleich lange und nur am Ende verschiedene Nummer repräsentiert werden und Unterbegriffe eine um eine Stelle verlängerte Nummer besitzen:

- 630 Agrarwirtschaft
  - 636 Tierhaltung
    - 636.7 Hunde
    - 636.8 Katzen

Da Dinge der realen Welt unter verschiedenen Aspekten betrachtet werden können, treten die sie bezeichnenden Begriffe in DDC mehrfach auf, beispielsweise befindet sich 'Kleidung' in 155.95 hinsichtlich ihres psychologischen Einflusses, in 391 in Bezug auf Brauchtum und in 746.92 als Aspekt der künstlerischen Gestaltung.

Gegenüber anderen Klassifizierungssystemen zeichnet sich DDC durch die weltweit hohe Verbreitung, häufige Aktualisierungen und vorhandene Übersetzungen in mehr als 30 Sprachen aus. Es existiert jedoch keine deutsche Übersetzung und die Verwendung von Nummern anstelle von Schlagwörtern erschwert eine ad hoc-Interpretation der Bezeichnungen einzelner Kategorien durch menschliche Nutzer.

Ein solches Klassifizierungssystem kann somit verwendet werden, um den Inhalt einer HTML-Seite oder einer vergleichbaren Ressource systematisch zu charakterisieren, was bei der Erstellung eines Nutzerprofils für die Ermittlung und Einordnung vorhandener Interessen verwendet werden kann, sofern es sinnvoll erscheint, diese aus den Seitenabrufen zu folgern.

Allgemeine Klassifizierungssysteme wie DDC, LCC oder UDC empfehlen sich besonders für den Einsatz bei Websites, die ein thematisch breites Spektrum von Inhalten abdecken, beispielsweise Nachrichtenportale oder Wissensdatenbanken, während bei sehr speziellem Content eine fachspezifische Klassifizierung wie MeSH für den medizinischen Bereich, eine individuelle Erweiterung einer bestehenden Klassifizierung oder eine vollkommen proprietäre Klassifizierung eingesetzt werden kann. Dabei könnte es sich im Falle eines Online-Shops um die bestehende Unterteilung des Sortiments in Warengruppen und einzelne Artikel handeln.

#### 3.4.1.3 Resource Description Framework

Im Frühjahr 2001 wurden unter der Bezeichnung 'Semantic Web Activity' alle Standardisierungsbemühungen des W3C zusammengefasst, die darauf abzielen, WWW-Ressourcen hinsichtlich ihres Inhalts, Kontexts, Urhebererschaft und weiterer Aspekte besser beschreiben zu können. Hierzu zählt insbesondere das Resource Description Framework (RDF), welches eine Reihe verschiedener Vorschläge integriert, Webressourcen mit maschinenlesbaren Metainformationen anzureichern.

"RDF can be used in a variety of application areas; for example: in resource discovery to provide better search engine capabilities, in cataloging for describing the content and content relationships available at a particular Web site, page, or digital library, (...) in content rating, (...) for describing intellectual property rights of Web pages, and for expressing the privacy preferences of a user as well as the privacy policies of a Web site" [LaSw99].

In dieser Spezifikation wird dabei versucht, die technischen Möglichkeiten der Beschreibung von Metainformationen einer Webressource unabhängig von der Anwendungsdomäne festzulegen. Dazu werden die Ressourcen (Resources) mittels sogenannter Statements beschrieben, in denen zuvor definierte Eigenschaften (Properties) der Ressource mit konkreten Werten belegt werden.

Zur Identifizierung einer Ressource wird deren URI benutzt, im Web also i. Allg. eine URL, während die Statements in XML ausgedrückt werden, wobei der verwendete XML-Namespace die zur Verfügung stehenden Properties definiert. Hierfür sieht RDF keinen bestimmten Namespace vor, legt jedoch nahe, vorhandene XML-basierte Standards zu nutzen, insbesondere DC für semantische Informationen sowie P3P für Privacy-Informationen.

Diese Integration geschieht durch die Angabe der XML-Namespace zu Beginn eines RDF-Statements. Folgendes Beispiel aus [LaSw99] veranschaulicht die Verwendung von DC zur Beschreibung von Metainformationen über die Website "www.dlib.org" mittels RDF:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/metadata/dublin_core#">
  <rdf:Description about="http://www.dlib.org">
    <dc>Title>D-Lib Program - Research in Digital Libraries</dc>Title>
    <dc>Description>The D-Lib program supports the community of people
      with research interests in digital libraries and electronic
      publishing.</dc>Description>
    <dc:Publisher>Corporation For National Research
      Initiatives</dc:Publisher>
    <dc>Date>1995-01-07</dc>Date>
    <dc:Subject>
      <rdf:Bag>
        <rdf:li>Research; statistical methods</rdf:li>
        <rdf:li>Education, research, related topics</rdf:li>
        <rdf:li>Library use Studies</rdf:li>
      </rdf:Bag>
    </dc:Subject>
    <dc>Type>World Wide Web Home Page</dc>Type>
    <dc:Format>text/html</dc:Format>
    <dc:Language>en</dc:Language>
  </rdf:Description>
</rdf:RDF>
```

Die zweite `xmlns`-Angabe verweist auf den DC-Namespace, was die anschließende Verwendung der in Kapitel 3.4.1.2 eingeführten Elemente des DCMES zur Beschreibung der Ressource erlaubt, in diesem Beispiel also `<dc>Title>`, `<dc>Description>`, `<dc:Publisher>` und `<dc:Format>`.

Die aus dem XML-Standard übernommene Erweiterbarkeit derartiger Beschreibungsschemata ermöglicht sowohl eine weitere Verfeinerung, z. B. die Verwendung der DDC zur Belegung der Eigenschaft `<dc:Subject>` anstelle der hier verwendeten Auflistung von Schlagwörtern, als auch die Hinzunahme anderer Metainformationen, z. B. der für diese Website gültigen Privacy-Policy gemäß P3P.



Die Assoziation von RDF-Statements zu Ressourcen kann nach [LaSw99] auf verschiedene Weisen erfolgen, im Fall von Websites oder –seiten ist insbesondere eine Einbettung der XML-Informationen in den HTML-Code denkbar.

#### 3.4.2 Zuordnung von Metainformationen zu statischem HTML

Die Anwendung der oben beschriebenen Möglichkeiten, Webressourcen mit Metadaten anzureichern und inhaltlich zu klassifizieren, erbringt einen Nutzen, wenn es zur Gewinnung von Profilinginformationen notwendig wird, die beobachteten Nutzerinteraktionen auf der Website auszuwerten. Die Lücke zwischen einer technisch exakten, aber betriebswirtschaftlich aussageschwachen Protokollierung von Verhalten und einer Anreicherung von Profilen um hinsichtlich der Unternehmensziele relevante Informationen kann auf diese Weise geschlossen werden.

Im technisch einfachsten Fall handelt es sich bei den Ressourcen einer Website um statische HTML-Seiten. Dies bedeutet, dass der redaktionelle Inhalt in einzelne Seiten fest encodiert ist und sich die Interaktionsmöglichkeiten eines Nutzers auf den gezielten Abruf dieser Seiten beschränken. Dennoch ist auch bei diesem geringen Maß des Informationsaustauschs die Erstellung individueller Nutzerprofile möglich und hilfreich, da sich auf diese Weise typische Nutzergruppen, deren Schwerpunktinteressen und Zusammenhänge zwischen diesen und anderen Merkmalen feststellen lassen.

Während die sogenannten 'hard facts' eines Nutzers, wie Name, Anschrift, Beruf, Einkommen usw. nur durch explizite Profildatengewinnung ermittelt werden können, ist es sinnvoll, die ebenfalls sehr interessanten 'soft facts', insbesondere Interessen und Präferenzen, aus dem Navigationsverhalten zu schließen, da aus der Benutzung eines Links und anschließender Betrachtung einer Seite i. Allg. ein Interesse an dort hinterlegten Inhalten gefolgert werden kann. Profilerweiterungen bei Attributen wie 'Hat Interesse an' sind deshalb immer möglich, wenn die abgerufene Seite in systematischer Weise semantisch klassifiziert ist und eine Abbildung dieser Klassen auf die in Profilschemata möglichen Attribute bzw. eine Relation zwischen diesen Mengen existiert.

Ein System für Profiling muss einen Seitenabruf gemäß dieser Zuordnung verarbeiten, indem es eine entsprechende Erweiterung des korrespondierenden Profils vornimmt. Die Zuordnung kann mithilfe einer Tabelle oder einer anderen Dictionary-Datenstruktur verwaltet werden, wobei sich für die eindeutige Identifizierung der Webresource deren URL anbietet, der ein oder mehrere Schlüssel zugewiesen werden, die Elemente des verwendeten Klassifikationssystems repräsentieren.

Der Aufbau dieser Zuordnungstabelle sollte systemseitig unterstützt bzw. im Idealfall automatisiert geschehen. Zur Identifikation aller verfügbaren HTML-Seiten kann ein hierfür vorgesehenes Werkzeug entweder das Dateisystem des Webserver durchsuchen oder per HTTP-Requests beginnend von der Einstiegsseite der Website alle eingebetteten Links verfolgen.

Die kategorisierenden Informationen können im einfachsten Fall mithilfe von selbst definierten HTML-Metatags direkt in die einzelnen Seiten hineincodiert werden, sodass durch

Parsen der Dateien die notwendige Zuordnung einer Semantik zu einer URL einfach zu erreichen ist.

Alternativ kann jedoch auch DC oder RDF verwendet werden sowie die bei diesen Standards zusätzlich vorgesehenen Möglichkeiten zur Ablage der Metadaten. So werden auf der Website der DCMI (<http://dublincore.org>) die zu jeder HTML-Datei gehörenden Metainformationen in einer XML-Datei abgelegt, deren ansonsten gleich lautende URL um '.rdf' verlängert wurde.

Natürlich erfordert die automatische Zuweisung einer Semantik zu einer URL eine zuvor erfolgte Einordnung des jeweiligen Inhalts in das verwendete Klassifizierungsschema. Diese Aufgabe muss i. Allg. von einem Redakteur durchgeführt werden und führt zu großem Aufwand, wenn ein derartiges Profilingssystem bei einer Website eingesetzt werden soll, die bereits eine große Menge Content enthält. Sofern ein CMS eingesetzt wird, sollte dies die semantische Klassifizierung neu eingestellter Inhalte nach Möglichkeit unterstützen, beispielsweise indem zu vergebende Keywords gemäß der für über- oder untergeordnete Seiten verwendeten Keywords vorgeschlagen werden oder eine Online-Publizierung ohne erfolgte Klassifizierung nicht möglich ist. Dabei sollte jedoch bedacht werden, dass eine sinnvolle Zuweisung einer konkreten Bedeutung zu einer vorhandenen URL nicht immer möglich ist, so existieren in aller Regel Webseiten, die den Nutzern die Navigation erleichtern, ohne dass aus deren Betrachtung ein bestimmtes Interesse gefolgert werden kann.

Zur Bewertung großer Textmengen, wie z. B. durch Mitglieder einer Community eingestellte Inhalte, kann ggf. Software aus dem Bereich der KI eingesetzt werden, welche durch Volltextsuche und unter Verwendung eines Thesaurus eine Klassifizierung erzeugen kann.

#### 3.4.3 Zuordnung von Metainformationen zu dynamisch erzeugtem HTML

Aufgrund der wachsenden Anforderungen an Aktualität und Variabilität bei kommerziellen Websites sind heutzutage kaum noch nur statische HTML-Seiten anzutreffen. Vielmehr werden CMS oder eigenentwickelte Anwendungen eingesetzt, die aus bereitgestellten Inhalten unter Verwendung gestalteter HTML-Templates Webseiten dynamisch generieren und dabei den anzuzeigenden Content aus einer Datenbank auslesen.

Eine typische URL zur Anzeige eines Produkts eines Online-Shops könnte beispielsweise wie folgt aussehen:

```
www.p-buy.de/catalog.jhtml?action=showitem&cat_id=472&product_id=015926
```

Dabei ist die generische Seite `catalog.jhtml` in der Lage, verschiedene Arten von Ergebnissen zu produzieren, in diesem Fall etwa die Ansicht eines bestimmten Produkts, das über eine Warengruppen- (`cat_id`) und Artikelnummer (`product_id`) referenziert wird.

Auch in diesen Fällen ist die Zuordnung einer Semantik zu der von einem Nutzer abgerufenen URL notwendig, handelt es sich doch um eine Interaktion, die beispielsweise ein deutliches Interesse eines potenziellen Konsumenten an einem Produkt belegt. Zur exakten Differenzierung zwischen den einzelnen Produkten, die ebenfalls erforderlich ist, muss nun jedoch eine Verarbeitung der gesamten URL einschließlich der per GET oder POST mitgelieferten Attribute erfolgen. Damit unterscheidet sich diese Situation grundsätzlich vom Abruf einer statischen HTML-Seite, insbesondere da die potenzielle Anzahl der

verschiedenen URLs sehr groß sein kann und die dynamische Generierung der Seiten eine persistente Ablage produktspezifischer Metainformationen im Dateisystem des Servers nicht sinnvoll macht.

Als Lösung für diese Problemstellung wird vorgeschlagen, die abgerufene URL zu parsen und die Pfadangabe in Verbindung mit den variablen Daten der Attribut-Werte-Paare auszuwerten. Die Bedeutung dieser Werte kann in einer oder mehreren separaten Tabellen hinterlegt werden, sofern die Werte nicht bereits selbst das Klassifizierungssystem bilden.

Für die Fortführung des oben begonnenen Beispiels heißt dies, nach dem Lesen der Pfadangabe `www.p-buy.de/catalog.jhtml` im Rest der URL nach allen möglicherweise vorkommenden Attributen zu suchen und die diesen zugewiesenen Werte zu interpretieren. So sind als mögliche Aktionen (`action`) neben `showitem` für die Ansicht eines Produkts auch andere denkbar, z. B. für das Anzeigen von Detailinformationen oder das Hinzufügen zum Warenkorb, wonach das Maß des Interesses an dem ausgewählten Artikel gefolgert werden kann. Somit können für die durch `cat_id` bestimmte Warengruppe und das durch `product_id` identifizierte Produkt entsprechende Zählvariablen im Profil des aktiven Nutzers erhöht werden bzw. für `cat_id` und `product_id` über weitere Tabellen Klassifizierungsschlüssel ermittelt werden (`cat_id=472` → 'Autoradios'), mit denen das Profil erweitert werden kann.

Zur Anwendung dieses Verfahrens bei existierenden Websites ließe sich die erstmalige Zuordnung von abgerufenen URLs und darin enthaltener Variablen auf durchzuführende Modifikationen von Profilen mithilfe eines Administrationswerkzeugs vereinfachen, das sowohl HTTP-Zugriff auf die Website als auch Datenbankzugriff auf Contenttabellen und das verwendete Profilschema hat.

Die Registrierung und Auswertung des Downloads einzelner Grafik-, Audio- und sonstiger Dateien kann auf die gleiche Weise geschehen, sofern diese zuvor kategorisiert wurden und die zugehörigen URLs in einer geeigneten Datenstruktur vorgehalten werden.

#### 3.4.4 Zuordnung von Metainformationen zu Transaktionen

Zur vollständigen Abdeckung aller für e-Business-Unternehmen relevanten Interaktionen auf HTML-basierten Websites müssen abschließend Transaktionen gesondert betrachtet werden. Darunter sollen nun Abfolgen von Seitenabrufen verstanden werden, die unbedingt zusammenhängend geschehen und abgeschlossen sein müssen, sich damit also den ACID-Eigenschaften von Datenbanktransaktionen nähern und in aller Regel einen geschäftlichen Austauschprozess widerspiegeln.

Das Standardbeispiel einer solchen Transaktion bildet der typische Einkauf in einem Online-Shop. Der eigentliche Einkaufsprozess beginnt mit der Auswahl des ersten Artikels, der in den virtuellen Warenkorb gelegt wird. Diese Aktion kann sich beliebig oft wiederholen und auch das Entfernen einzelner Artikel aus dem Warenkorb oder die Änderung der Bestellmenge sind i. d. R. möglich.

Dem 'Gang zur Kasse' entspricht in der Online-Welt das Auslösen der Bestellung des Warenkorb-inhalts, wobei in dieser Phase Name und Anschrift des Bestellers sowie die Lieferadresse, Versand- und Zahlungsart angegeben werden. Erst das finale Absenden der Bestellung schließt die Transaktion ab und löst damit die weitere Bearbeitung des Geschäftsvorgangs auf Seite des Anbieters aus.

Es ist zu beachten, dass gut realisierte Online-Bestellvorgänge dem Nutzer das zwischenzeitliche Betrachten weiterer Artikel oder gar anderer Webseiten im gleichen Browserfenster erlauben, ohne Informationen über den aktuellen Fortschritt der Bestellung zu verlieren.

Zur Abbildung und Registrierung derartiger Abläufe muss ein Profilingssystem die auf einer Website möglichen Transaktionen kennen und verfolgen können. Dazu wird vorgeschlagen, Transaktionen in einem Referenzmodell für Profiling gesondert zu betrachten und beginnend mit den potenziellen Startpunkten eines solchen Ablaufs dessen Voranschreiten mittels geeigneter Mechanismen zu überwachen. Im Gegensatz zur ausschließlichen Registrierung der die Transaktion abschließenden Aktion kann ein Profilingssystem auf diese Weise protokollieren, welche Daten innerhalb der Transaktion ausgetauscht wurden (bestellte Waren, übermittelte Angaben), ohne von der direkt beteiligten Webapplikation explizit darüber informiert zu werden. Somit können auch abgebrochene Transaktionen mit der Kenntnis ihres letzten Zustands eindeutig individuellen Nutzerprofilen zugeordnet werden (Problem der 'abandoned shopping carts').

## 4 Ein Modell für Profiling

In diesem Kapitel erfolgt die Erstellung eines Modells für Profiling bei e-Business-Websites. Die grundsätzliche Vorgehensweise besteht dabei in der Ermittlung der zentralen Aufgaben eines Profilingssystems in Form von Anwendungsfällen sowie der zu deren Abwicklung benötigten grundlegenden Objekte.

Das Ergebnis dieser Analyse wird unter Berücksichtigung der in Kapitel 2 definierten Anforderungen und unter Verwendung der in Kapitel 3 beschriebenen technischen Grundlagen zu einem Grobentwurf für ein Profilingssystem erweitert.

Dabei kommen Diagramme der *Unified Modeling Language* (UML) zum Einsatz, deren aktuelle Version 1.3 in [OMG00] spezifiziert ist, und die durch textuelle Beschreibungen ergänzt werden.

### 4.1 Objektorientierte Analyse

Zunächst wird analysiert, welche Anwendungsfälle für ein Profilingssystem existieren, sowie welche Entitäten der realen Welt als elementare Objekte in einem solchen System abgebildet werden müssen.

Das Ergebnis dieser Analyse ist also ein Modell dessen, was im Umfeld der gegebenen Problemstellung vorgefunden wird, und entspricht damit am ehesten dem, was [FoSc00] als "konzeptionelle" Sichtweise bezeichnet. "Wenn man eine konzeptionelle Sichtweise einnimmt, erstellt man ein Diagramm, das die Konzepte im untersuchten Problembereich wiedergibt" [FoSc00].

#### 4.1.1 Anwendungsfälle

Zur Ergründung der wesentlichen Aufgaben eines Profilingssystems werden zunächst verschiedene Anwendungsfälle hierfür entwickelt. Da ein Anwendungsfall im Sinne der UML eine für viele Zwecke zu grobe Beschreibung darstellt, wird eine weitere Unterteilung in Szenarien vorgenommen, wie [FoSc00] sie vorgeschlägt: "Ein Szenario ist eine Folge von Schritten, die die Interaktion zwischen einem Benutzer und einem System beschreibt. (...) Ein Anwendungsfall ist eine Menge von Szenarien, die durch ein gemeinsames Benutzerziel verbunden sind".

Die grundlegenden Anwendungsfälle für ein Profilingsystem für Websites bestehen aus der Erfassung und Protokollierung der Nutzerinteraktionen sowie der daraus resultierenden Erweiterung des Nutzerprofils.

Aus den in Kapitel 3.4 genannten technischen Gründen wird in Abbildung 7 bereits zwischen

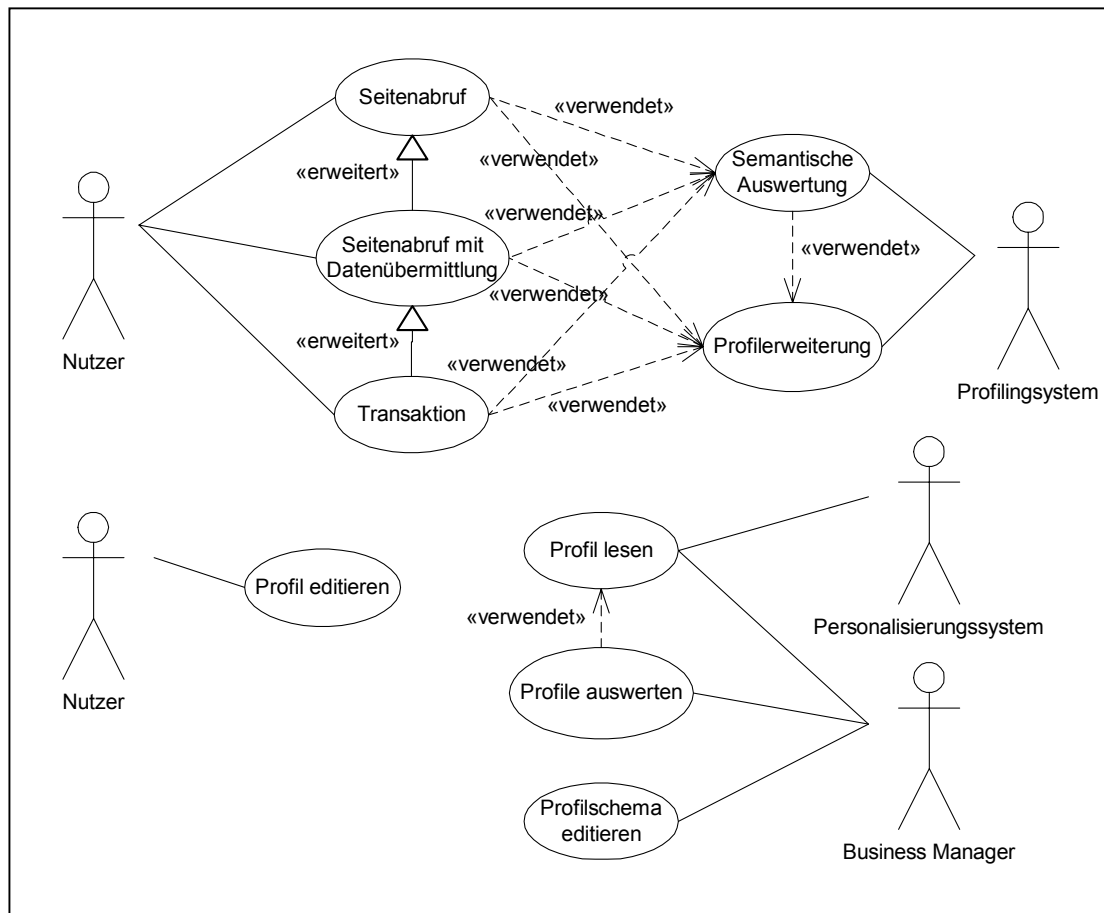


Abbildung 7: Anwendungsfalldiagramm Profiling

den Anwendungsfällen 'Seitenabruf' als einfache Übertragung einer HTML-Seite zum Nutzer, 'Seitenabruf mit Datenübertragung' wie dem Ausfüllen und Absenden eines HTML-Formulars sowie 'Transaktion' unterschieden, wobei es sich i. d. R. um einen 'Seitenabruf mit Datenübertragung' innerhalb eines bestimmten Kontexts handelt. Diese stufenweise Anreicherung der Anwendungsfälle um Interaktionskomplexität kommt in der Anwendung der 'erweitert'-Beziehung zum Ausdruck.

Zur Veranschaulichung dieser Anwendungsfälle werden nachfolgend exemplarische Szenarien beschrieben, die als durchgängige Beispiele zur gedanklichen Evaluierung des nachfolgenden Entwurfs dienen sollen:

1. 'Seitenabruf':

- a. Ein Nutzer ruft eine aktuelle Nachrichtenmeldung auf einer Website ab.
- b. Ein Nutzer ruft einen redaktionellen Artikel auf einer Website ab.
- c. Ein Nutzer ruft eine ausführliche Produktbeschreibung innerhalb eines Online-Shops ab.

2. 'Seitenabruf mit Datenübertragung':

- a. Ein Nutzer bewertet einen redaktionellen Artikel einer Website ('sehr interessant', 'interessant', 'weniger interessant', 'uninteressant').
- b. Ein Nutzer teilt seinem Energieversorgungsunternehmen den Zählerstand des Stromzählers per Web-Formular mit.
- c. Ein Nutzer erstellt eine Nachricht für ein Bulletin-Board-System.

3. 'Transaktion':

- a. Ein Nutzer tätigt einen Online-Kauf mit folgenden Schritten:
  - Ansicht und Auswahl einer Menge von Artikeln
  - Eingabe der persönlichen Daten (Name, Anschrift, Zahlungs- und Versandinformationen)
  - Abschließende Bestätigung der Bestellung nach Angabe aller erforderlichen Daten
- b. Ein Nutzer tätigt eine Geldanlage gemäß Vorschlag eines Empfehlungssystems:
  - Authentifizierung durch Nutzerkennung und Passwort.
  - Interaktives Beantworten einer Reihe von Fragen zum Thema Geldanlage (Höhe des Anlagebetrags, Wertschätzung verschiedener Anlageformen, Risikobereitschaft u. ä.).
  - Auswahl und Abschluss einer Geldanlage aus einer Reihe nach Empfehlungsgrad sortierter Vorschläge.
  - Abschließende Bestätigung der Transaktion durch Eingabe der banküblichen Kennungen PIN und TAN.

Diese drei genannten Anwendungsfälle des Abrufs von Webseiten führen zu weiteren, vom Profilingssystem automatisch veranlassten Abläufen, die als eigene Anwendungsfälle modelliert werden können. Zunächst erfolgt die einfache Protokollierung des Seitenabrufs im Profil des Nutzers durch Speicherung der in Logfiles üblichen Informationen, es findet jedoch auch eine semantische Auswertung der Interaktion statt, deren Ergebnis ebenfalls das Nutzerprofil erweitert, indem beispielsweise ein Interesse oder eine Präferenz des Nutzers gefolgert und gespeichert wird.

Schließlich soll ein Profilingssystem die Möglichkeiten vorsehen, dass Nutzer bestimmte Teile ihres eigenen Profils editieren können und Systemanwender sowie andere Systeme auf Serverseite zumindest lesend auf die erstellten Profile zugreifen können.

#### 4.1.2 Grundlegende Objekte

In diesem Abschnitt erfolgt die Identifikation für das Profiling grundlegender Objekte<sup>6</sup>, die sich in natürlicher Weise aus der Betrachtung der oben geschilderten Anwendungsfälle und den daran beteiligten Akteuren ergeben.

<sup>6</sup> Der Begriff 'Objekt' bezeichnet in diesem Abschnitt genau genommen Objektklassen der realen Welt, wie 'Nutzer', 'Nutzerprofil' usw., welche in den UML-Diagrammen als Klassen modelliert werden. Erst das Erzeugen konkreter Instanzen dieser Klassen führt zu Objekten im sprachlichen Sinne der UML, die dann ggf. Entitäten der realen Welt 1:1 repräsentieren.

## 4.1.2.1 User

Das für ein Profilingssystem zentrale Objekt stellt der Nutzer einer Website dar, dessen Interaktion mit dem Online-Medium protokolliert wird und zu dem ein korrespondierendes Profil aufgebaut und erweitert werden soll.

Abbildung 8 spiegelt dies wider, indem je ein Objekt `User` ein Objekt `Session` und ein Objekt `Profile` aggregiert.

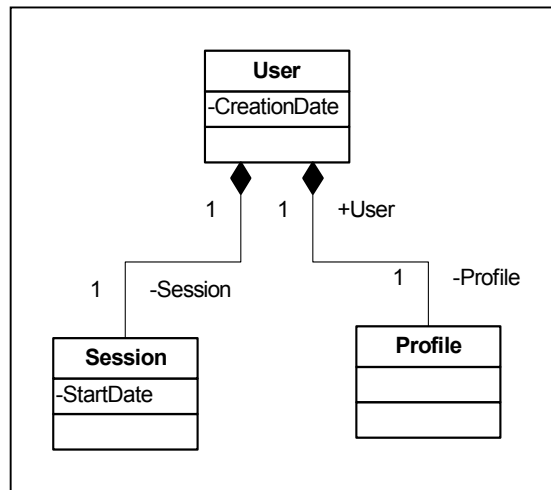


Abbildung 8: Klassendiagramm `User`

Dabei repräsentiert `Session` die aktuelle Session des Nutzers während der Interaktion mit der Website, welche nach deren Beendigung als Profilerweiterung gespeichert werden kann. `Profile` hingegen stellt das gesamte Nutzerprofil von `User` dar, welches bereits vor Beginn der aktuellen Session aufgrund eines früheren Besuchs bestanden haben kann und zur späteren Verwendung in gleicher Weise nach dem Ende der Session persistent gemacht werden sollte, falls eine Zuordnung gespeicherter Profile zu Nutzern möglich ist.

## 4.1.2.2 Request

Die Interaktion eines Nutzers mit einer Website lässt sich nach den Überlegungen aus Kapitel 3.1.1 als eine Folge von HTTP-Requests auffassen. Ein einzelner `Request` kann mit einem Zeitstempel versehen werden, besitzt eine HTTP-Methode und eine Pfadangabe der angeforderten URL.

Darüber hinaus kann ein `Request` Header-Informationen (`Header`) und Nutzerdaten (`UserData`) enthalten, wobei letztere in verschiedenen Ausprägungen vorliegen können.



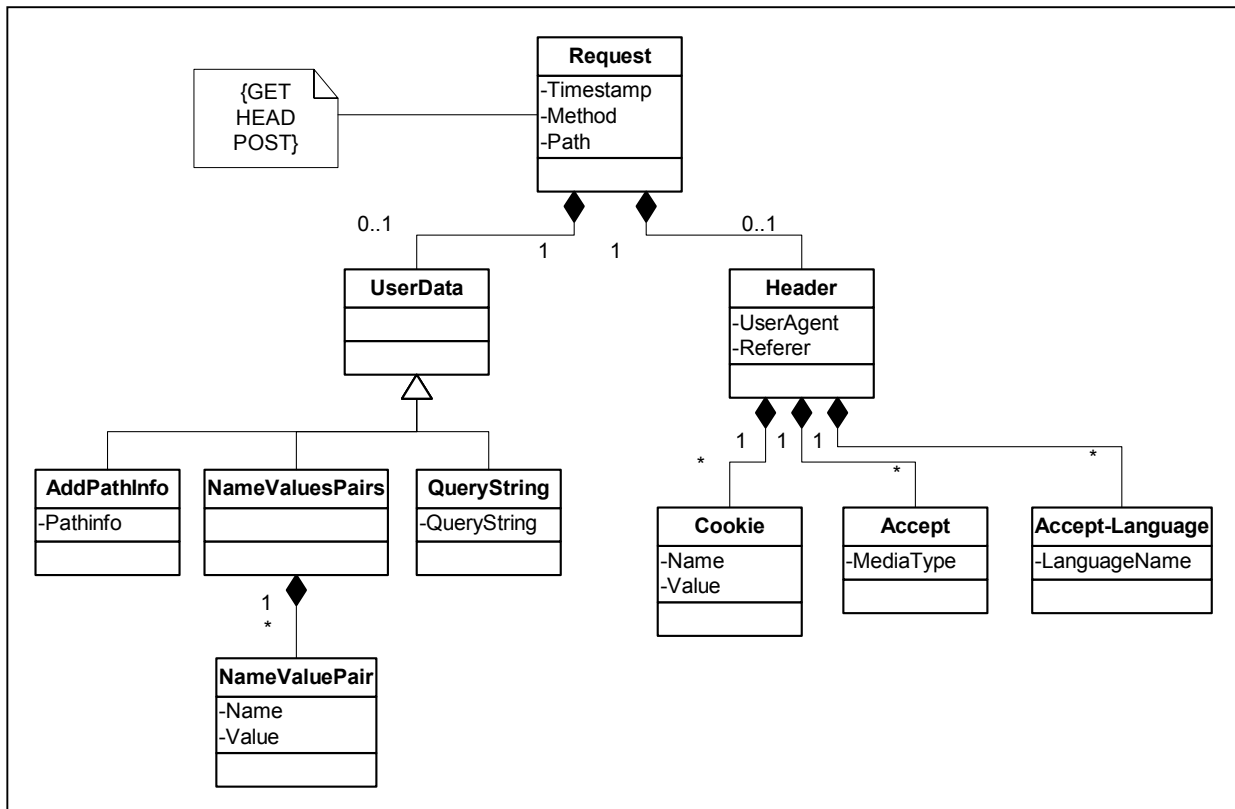


Abbildung 9: Klassendiagramm Request

In Abbildung 9 wird zunächst zwischen einer zusätzlichen Pfadangabe (*AddPathInfo*), einer Folge von Attributnamen-Werte-Paaren (*NameValuePairs*) oder einer beliebig formatierten Zeichenkette (*QueryString*) unterschieden, wie sie beispielsweise bei der Übermittlung von Suchanfragen verwendet wird.

#### 4.1.2.3 Sessions und Transactions

Eine zusammenhängende Folge zeitlich geordneter *Requests* eines einzelnen Nutzers bei einer Website wird als *Session* bezeichnet und soll im Sinne eines Besuchs dieser Website verstanden werden.

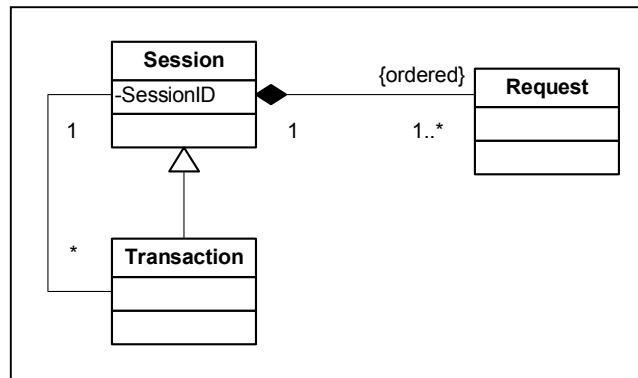


Abbildung 10: Klassendiagramm Session - Transaction

Abbildung 10 verdeutlicht dabei auch, dass Transaktionen (Transactions) aus der analytischen Sicht als spezielle Form der Session verstanden werden können: während die Requests, die eine allgemeine Session bilden, lediglich nach ihrem Zeitstempel geordnet sind und beliebige URLs der Website enthalten, ist eine Transaction eine Folge bestimmter Requests, die dazu in einer transaktionsspezifischen Reihenfolge auftreten müssen. Es ist offenkundig, dass eine Session eine beliebige Menge Transactions enthalten kann.

#### 4.1.2.4 Profile

Neben der aktuellen Session modelliert Abbildung 8 vor allem das Nutzerprofil (Profile) als wesentliches Objekt eines Profilingsystems. Ein Vorschlag für den Aufbau eines solchen Profile und damit eine Spezifikation eines Basis-Profilschemas ist in Abbildung 11 dargestellt.

Dabei werden die Profildaten eines Nutzers in vier Hauptkategorien unterteilt:

- Basisdaten (BaseData): Diese enthalten grundlegende Angaben zur Person wie Name, Geburtsdatum, Geschlecht sowie Kontaktinformationen (Contact), die hier in Anlehnung an das Basisdatenschema von P3P aus [Cran00] modelliert wurden und sich in die Bereiche Post-, Telekommunikations- und Online-Kontaktinformationen untergliedern.
- Soziodemografische Daten (Sociodemographic): Hier werden Informationen über den Familienstand, das Einkommen, den Beruf, die Ausbildung u. ä. gehalten.
- Transaktionen (Transaction): Dieser Bereich enthält alle abgeschlossenen Transaktionen im o. g. Sinne, die zwischen Nutzer und Anbieter stattgefunden haben. Dabei ist ausdrücklich vorgesehen, neben den Online-Transaktionen auch alle Offline-Transaktionen hier zu speichern, um zu einer ganzheitlichen Sicht des Nutzers zu gelangen. Die Beziehung jeder Transaction zu einer Interaction (s. u.) hilft dabei, den Kontext dieses geschäftlichen Austauschprozesses zu erhalten. Je nach Anwendungsdomäne des Profilings müssen an dieser Stelle neben Käufen oder Bestellungen auch andere Transaktionsarten berücksichtigt werden.
- Interaktionen (Interaction): Hier werden Berührungen zwischen Anbieter und Nutzer gespeichert, die nicht zwangsläufig eine Transaktion enthalten müssen. So handelt es sich bei jeder WWWSession um eine Interaction, die gespeichert werden sollte, sofern sie einem bestimmten Nutzer zugeordnet werden kann; nach

den Überlegungen aus Kapitel 2.1.4 empfiehlt sich aber auch das Speichern telefonischer, postalischer oder persönlicher Interaktionen.

- Präferenzdaten (PreferenceData): Diese enthalten die aus den Interactions gewonnenen Informationen über Interessen und Vorlieben eines Nutzers, also den aus Marketingsicht besonders relevanten Daten, die den eigentlichen Gewinn der Profilingmaßnahmen ausmachen und als Basis für eine Personalisierung der Geschäftsbeziehung dienen können. Dieser Bereich wird zunächst nicht näher spezifiziert, da seine genaue Ausgestaltung unmittelbar von der jeweiligen Anwendungsdomäne abhängt.

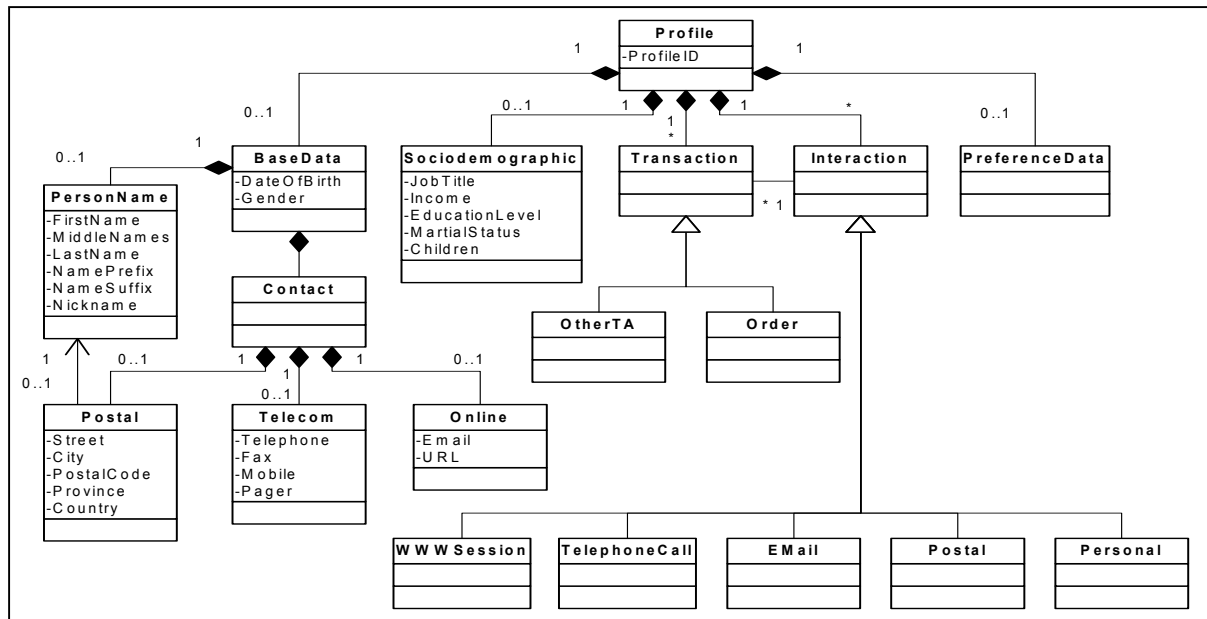


Abbildung 11: Klassendiagramm Profile

### **4.2 Entwurf eines Modells für Profiling**

Ziel dieses Abschnitts ist die Anreicherung und Modifikation der in Kapitel 4.1 gefundenen grundlegenden Objekte um weitere Objekte, die technisch notwendig oder sinnvoll sind, um die im Kapitel 2 aufgestellten funktionalen und nicht-funktionalen Anforderungen an ein Referenzmodell für Profiling zu realisieren. Diese Sichtweise wird in [FoSc00] als "spezifizierend" bezeichnet, wobei es sich bereits um einen Entwurf handelt, jedoch vorrangig Schnittstellen betrachtet werden und von Details der Implementierung abstrahiert wird.

Darüber hinaus wird neben der statischen Sicht auf das Modell das Zusammenspiel der enthaltenen Elemente beschrieben, wodurch auch Abläufe und damit die Funktionsweise des Systems in seiner Gesamtheit ersichtlich werden.

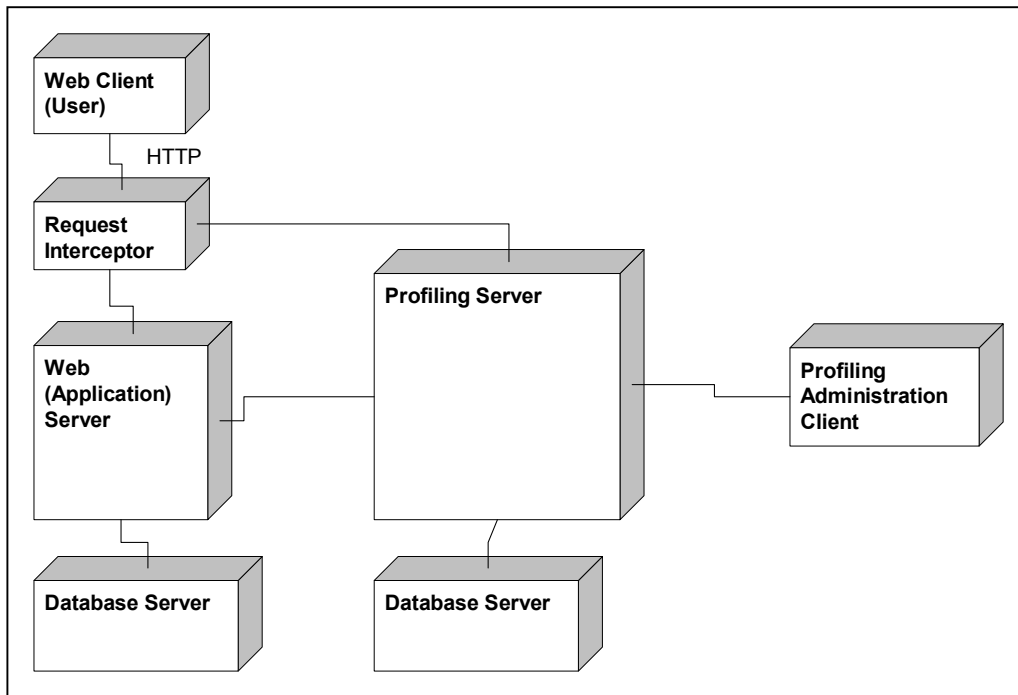
#### 4.2.1 Unterstützende Objekte und Objektverteilung

Zunächst erfolgt die Modellierung zusätzlicher, technisch notwendiger Objekte sowie deren möglicher Verteilung in einem Gesamtsystem aus Client, Server und Profilingssystem.

##### 4.2.1.1 Request-Interceptor

Aufgrund der als sehr wichtig einzustufenden Anforderung der einfachen Integrationsmöglichkeit in bestehende und sich häufig ändernde Internetanwendungsumgebungen, wie sie in Kapitel 2.5.3 diskutiert wurde, und basierend auf den technischen Kommunikationsgrundlagen, wie sie in Kapitel 3.1 vorgestellt wurden, wird vorgeschlagen, das Profiling von der eigentlichen Internetanwendung zu trennen und als externalisiertes System oder Komponente zu entwerfen. Damit wird auch im Sinne des Referenzcharakters dieses Modells eine größtmögliche Unabhängigkeit von einer tatsächlichen Hardware/Software-Umgebung eines Anwendungsbeispiels erreicht.

Aus technischer Sicht ist dies durch das einfache 'Abhören' der Client-Server-Kommunikation einer Webanwendung möglich; eine hier 'Request Interceptor' genannte Komponente kann alle Anfragen von Nutzern einer Website duplizieren und an eine Profilingkomponente weiterleiten, die das Verhalten der Nutzer prinzipiell unabhängig und parallel zur Webanwendung verarbeiten und zur Profilgewinnung nutzen kann. Ein hohes Maß der Verteilung der verschiedenen Subsysteme bei dieser Vorgehensweise ist in Abbildung 12 dargestellt.



**Abbildung 12: Verteilungsdiagramm Profiling**

Der Request Interceptor ist dem Web-Client und Webserver zwischengeschaltet und empfängt die HTTP-Requests der Nutzer, die er sowohl an den Webserver als auch an das Profilingssystem weiterleitet. Der Webserver beantwortet die Anfrage des Clients in gewohnter Weise, wobei es sich um einen einfachen Webserver zur Auslieferung statischer HTML-Seiten, aber auch um einen Applikationsserver handeln kann, welcher auf einer Datenbank basierend die Abwicklung komplexer Geschäftsprozesse per WWW ermöglicht.

Dies impliziert, dass die Realisierung des Request Interceptors zwar stark von der Systemumgebung der Zielanwendung abhängt, bei einer wohlbedachten Implementierung auf die Schnittstelle des Profilingsystems hin dieses jedoch einheitlich für eine große Vielfalt möglicher Webanwendungen einsetzbar ist. Im einfachsten Fall könnte ein Request Interceptor als Shell- oder Perl-Skript realisiert werden, performanter und leichter integrierbar ist bei Verwendung eines Webserver jedoch die Nutzung des zur Verfügung gestellten API, z. B. NSAPI oder ISAPI. Applikationsserver bieten häufig eine vereinfachte Möglichkeit zum Abspalten von Requests, erlauben aber auch immer den Einsatz einer eigenentwickelten Komponente, die der eigentlichen Requestverarbeitung vorgeschaltet wird und die das Profilingssystem mit den erforderlichen Daten ansteuert. Falls der Applikationsserver bereits mit der Vergabe von Session-IDs arbeitet, können diese in beiden Systemen gemeinsam verwendet werden und so den Aufwand der Nutzeridentifikation für einen Datenaustausch zwischen den Systemen auf ein Minimum reduzieren, was einer Personalisierung durch den Applikationsserver auf Basis der Profildaten des Profilingsystems entgegenkommt. Hierzu dient die Verbindung zwischen 'Web (Application) Server' und 'Profiling Server' in Abbildung 12.

In den zuletzt aufgeführten Fällen der Nutzung eines Server-API zur Abspaltung des Client-Requests liegt keine Verteilung des Request Interceptors auf einen dedizierten Netzknotten oder Rechner vor, wie sie in obiger Abbildung dargestellt ist; diese ist jedoch möglich, um

beispielsweise die in Kapitel 3.2.3 beschriebenen Techniken der Lastverteilung zu nutzen, bei denen Anfragen reihum oder gemäß freier Systemressourcen auf verschiedene Web- oder Applikationsserver verteilt werden. Auf diese Weise kann das Profiling für mehrere Webserver an einen Profilingserver delegiert werden oder auch mehrere Profilingserver fest oder variabel mehreren Webservern zugeordnet werden.

Die Kommunikation zwischen den Knoten geschieht mit nahezu beliebigen Mitteln: während sich für die Weiterreichung der Client-Requests an den Server der eigentlichen Webanwendung ebenfalls HTTP empfiehlt, kann für den übrigen Informationsaustausch je nach Realisierungstechnologie RMI, CORBA oder sonstige Middleware eingesetzt werden.

Bei einer Nutzung der Profilingkomponente in Verbindung mit einem einfachen Webserver kann eine sichere Unterscheidung einzelner Nutzer zur Generierung individueller Profile wegen der in Kapitel 3.1.3 diskutierten Probleme nur sichergestellt werden, wenn der Request Interceptor das aufwändige URL-Rewriting beherrscht und damit ein Session-Management implementiert.

Die vollständige Auslagerung des Profilings aus der Webapplikation, auf die es angewendet wird, bringt den Vorteil mit sich, dass Nutzerprofile unabhängig von der Art der Webapplikation und sogar für Webserver statischer HTML-Seiten angelegt werden können. Durch die klare Trennung von Web- und Profilingserver und die Verwendung schlanker Schnittstellen zwischen diesen, über welche vor allem URLs einschließlich angehängter Nutzerdaten übertragen werden, ist eine umfassende Modifikation und sogar ein Austausch der Webanwendung unter Beibehaltung des Profilingsystems und der darin abgelegten Profile denkbar.

Der entscheidende Nachteil dieser Vorgehensweise ist die Notwendigkeit, zur Interpretation des Nutzerverhaltens anhand der vom Client angefragten URLs die Applikationslogik der Webanwendung im Profilingssystem größtenteils nachbilden zu müssen. Dieser nicht unerhebliche Aufwand kann jedoch durch geeignete Hilfsmittel unterstützt und durch die Verwendung einer gemeinsamen Datenbank reduziert werden, wie in den nachfolgenden Abschnitten noch gezeigt wird.

### 4.2.1.2 SessionManager und ProfileManager

Die in Kapitel 4.1.2 identifizierten grundlegenden Objekte<sup>7</sup> `Session` und `Profile` repräsentieren die beiden zentralen Entitäten, die einem einzelnen Nutzer einer Website zuzuordnen sind. Während `Session` temporär die Sitzung eines Nutzers beschreibt, enthält `Profile` alle dauerhaft zu speichernden Informationen, auf die während einer Sitzung effizient zugegriffen werden können sollte, die aber auch zwischen verschiedenen Sitzungen verfügbar sein müssen.

---

<sup>7</sup> An dieser Stelle findet zugunsten einer einfacheren Lesbarkeit ein Wechsel in der Bedeutung des Begriffs 'Objekt' statt: nachdem Entitätenklassen der realen Welt wie im vorangegangenen Abschnitt als UML-Klassen modelliert werden, bezeichnet 'Objekt' nachfolgend Instanzen dieser Klassen, da überwiegend deren konkrete Interaktionen beschrieben werden. Dies trifft auch bei ausschließlicher Verwendung des Klassennamens für ein Objekt zu, was – grundlegende Kenntnisse objektorientierter Konzepte vorausgesetzt – auch aus dem jeweiligen Kontext hervorgeht.

Zur Regelung des Zugriffs auf diese nutzerindividuellen Entitäten wird die Einführung eines `SessionManagers` und eines `ProfileManagers` vorgeschlagen, die innerhalb der Profilingkomponente in der in Abbildung 13 gezeigten Beziehung zueinander stehen.

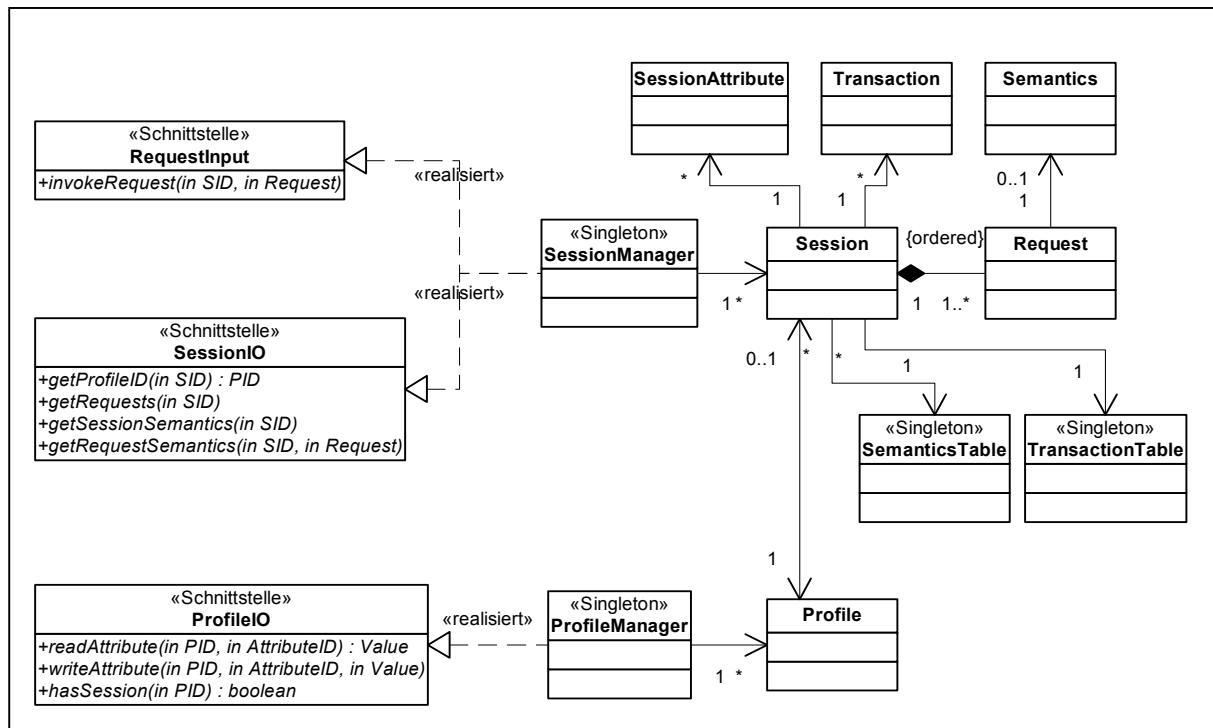


Abbildung 13: Klassendiagramm Sessionmanagement

Der `SessionManager` realisiert insbesondere die Zuordnung eingehender `Requests` zu den vorhandenen `Sessions`, indem er die Schnittstelle `RequestInput` implementiert, die vom oben eingeführten `Request Interceptor` durch Aufruf der Methode `invokeRequest()` angesprochen wird. Eine einheitliche `Session-ID (SID)` dient dabei der durchgängigen Identifizierung einzelner Nutzer, die sich zu einem Zeitpunkt auf der Website befinden.

Jede `Session` besteht aus einer Folge einzelner `Requests`, welche zu ihr gehören und die in der Reihenfolge ihres Eintreffens geordnet sind. Sofern dem `Request` in der `SemanticsTable` eine Bedeutung (`Semantics`) zugeordnet war, wird diese vom korrespondierenden `Request`-Objekt fortan referenziert.

`Requests`, die zu Transaktionen gehören, werden durch Nachschlagen in der `TransactionTable` erkannt und führen zu einer Weiterverarbeitung des `Requests` durch einen endlichen Automaten, der das Voranschreiten der Transaktion überwacht, wozu pro Transaktionstyp ein `Transaction`-Objekt an die jeweilige `Session` gebunden wird.

Zusätzlich besteht die Möglichkeit, mittels `SessionAttribute` beliebige Einzelinformationen im Sinne einer sitzungsabhängigen Merkvariablen an `Session` zu knüpfen, um einfache, den Kommunikationskontext kennzeichnende Attribut-Werte-Paare verwenden zu können.

Zum Zugriff auf diese `Session`-relevanten Informationen implementiert der `SessionManager` ebenfalls die Schnittstelle `SessionIO`, über die alle `Requests` einer

`Session` erfragt werden können (`getRequests()`), die dazugehörigen Einordnungsschlüssel der semantischen Klassifikation (`getSessionSemantics()`) sowie der Einordnungsschlüssel eines bestimmten Requests (`getRequestSemantics()`).

Besonders wichtig ist bei dieser Schnittstelle die Methode `getProfileID()`, welche die Identifizierung des zu einer Sitzung gehörenden Nutzerprofils anhand der Session-ID erlaubt. Dieses Profil besteht in jedem Fall, da es bei Wiederholungsbesuchern einer Website unmittelbar nach deren Identifizierung aus dem Persistenzspeicher gelesen wird und bei neuen oder noch nicht identifizierten Nutzern implizit zu Beginn der Session angelegt wird, um den in Kapitel 2.1.1 aufgestellten Anforderungen gerecht zu werden.

Die Schnittstelle `ProfileIO` bietet den Zugriff auf einzelne Profile, der vom `ProfileManager` realisiert wird. Die Methoden `readAttribute()` und `writeAttribute()` erlauben das Auslesen und Beschreiben einzelner Attribute gemäß dem zugrunde liegenden Profilschema. Da dies auch bei Profilen möglich ist, deren Nutzer zur gleichen Zeit nicht online ist, bietet `hasSession()` die Möglichkeit, diesen Sachverhalt zu überprüfen.

Die Verteilung der gebotenen Funktionalität auf drei Schnittstellen geschieht, um jeweils "eine einfache Schnittstelle zu einem komplexen Subsystem anbieten" zu können [GHJV96]. Damit wird die Motivation des Entwurfsmusters 'Fassade' aus [GHJV96] aufgegriffen. Während `RequestImport` ausschließlich für die Verarbeitung der vom `Request Interceptor` übermittelten HTTP-Anfragen bestimmt ist, bündelt `SessionIO` alle Zugriffsmöglichkeiten auf Daten der bestehenden Sitzungen, was insbesondere für die Personalisierung zur Laufzeit des Gesamtsystems genutzt werden kann. `ProfileIO` erlaubt schließlich das Lesen und Schreiben von Profilen unabhängig von korrespondierenden Sessions, sodass diese Schnittstelle allein für den Datenzugriff seitens weiterer Analysekomponenten ausreichend ist. "Die Einführung einer Fassade entkoppelt die Subsysteme von Klienten und anderen Subsystemen, wobei die Unabhängigkeit und Portabilität des Subsystems gefördert wird" [GHJV96].

Aufgrund der Koordinationsaufgabe von `SessionManager` und `ProfileManager` empfiehlt sich eine Realisierung als 'Singleton', womit sichergestellt wird, dass zur Laufzeit des Systems jeweils maximal eine Instanz dieser Klassen existiert. Dieses Vorgehen ist auch für `SemanticsTable` und `TransactionTable` sinnvoll.

### 4.2.1.3 SemanticsTable, TransactionTable und Classification

Die Objekte `SemanticsTable` und `TransactionTable` realisieren Dictionary-Datenstrukturen. `SemanticsTable` erlaubt insbesondere das Zuordnen verschiedener Schlüssel zu einer gegebenen URL gemäß der zugrunde liegenden semantischen Klassifikation, von denen einige beispielhaft in Kapitel 3.4.1 vorgestellt wurden.



Die Modellierung einer solchen Klassifikation ist dem Paket `Classification` der Abbildung 14 zu entnehmen. Dieses enthält im Wesentlichen das Entwurfsmuster 'Kompositum', bestehend aus `Component`, `Composition` und `Leaf`, welches in [GHJV96] vorgestellt und in [Vlis99] erläutert wird. Dieses Muster "definiert Klassenhierarchien auf Basis von Klassen für primitive und zusammengesetzte Objekte" [GHJV96] und eignet sich damit gut zur Repräsentation hierarchischer Strukturen wie der Dewey Decimal Classification oder der Universal Decimal Classification. In Verbindung mit einem Bezeichner für das jeweils verwendete Klassifikationssystem wird zudem das Analysemuster 'Identifikationsschema' realisiert. "Identifikationsschemata stellen den Kontext dar, in dem ein Objekt identifiziert wird" [Fowl99]. Somit können prinzipiell mehrere Klassifikationssysteme parallel verwendet werden.

Neben verschiedenen Semantik-Schlüsseln (`SemanticsElement`) können einer URL in

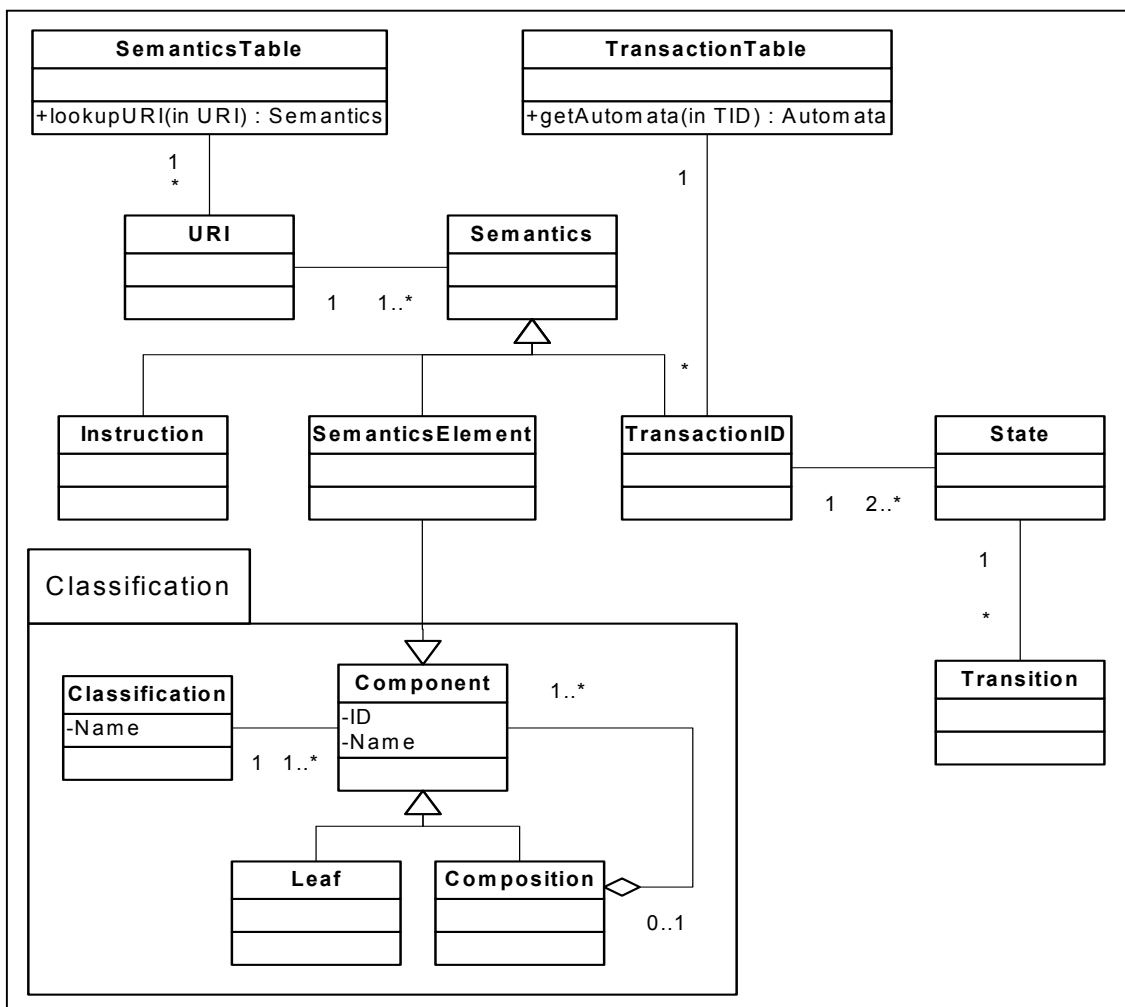


Abbildung 14: Klassendiagramm `SemanticsTable` und `TransactionTable`

`SemanticsTable` aber auch mehrere Anweisungen (`Instruction`) und eine Transaktions-ID zugewiesen sein. Letztgenannte impliziert, dass der erfolgte Seitenabruf Teil einer Transaktion ist, welche als solche überwacht werden muss. Dazu ist in `TransactionTable` pro Transaktionstyp die Struktur eines endlichen Automaten hinterlegt. Die Zustände des Automaten repräsentieren dabei den jeweils aktuellen Status der Transaktion, wobei Zustandsübergänge durch für diese Transaktion relevante HTTP-

Requests ausgelöst werden. Existiert zu einem Transaktionstyp noch keine Instanz des entsprechenden Überwachungsautomaten, so wird er mit dem Abruf der ersten URL, die zu dieser Transaktion gehört, aufgebaut und an das `Session`-Objekt des Nutzers gebunden.

Für jeden weiteren Transaktionsschritt genügt das Nachschlagen des Transaktionstyps in der `SemanticsTable`, da über die Transaktions-ID der bereits instanziierte Überwachungsautomat zur weiteren Verarbeitung des Requests angesprochen werden kann.

Endet die Transaktion erfolgreich, was durch die Geschäftslogik eindeutig definiert ist, so erreicht der Automat einen entsprechenden Endzustand und die Transaktion kann im Profil des Nutzers eingetragen werden. Nicht beendete Transaktionen werden am Ende der Session erkannt und können ebenfalls ausgewertet werden.

Dieses Vorgehen impliziert, dass Transaktionen des gleichen Typs bei demselben Nutzer nicht verschachtelt ablaufen können; diese Restriktion ist bei Websites i. d. R. ohnehin erwünscht.

Die als `Instruction` bezeichneten Anweisungen dienen dagegen der Durchführung einfacher Aktionen, die bei bestimmten Seitenabrufen zusätzlich stattfinden sollen und von denen folgende Grundtypen vorgeschlagen werden: `writeAttribute` und `readAttribute` zum Schreiben und Lesen von Attributen des Profils eines korrespondierenden Nutzers, `writeSessionAttribute` und `readSessionAttribute` für sitzungsbezogene Attribute, `createProfile` und `loadProfile` zum erstmaligen Erzeugen oder Laden des Nutzerprofils, sowie `stopSession` zum Beenden der Nutzersitzung. Die beiden letztgenannten Anweisungen werden insbesondere dann benötigt, wenn ein Request das Login oder Logout eines Nutzer indiziert.

Darüber hinaus ist dieser Mechanismus dazu geeignet, mittels eines API genutzt zu werden, um eine Integration zu anderen Systemen zu schaffen, beispielsweise einen automatisierten Faxversand zu starten oder über ein Call Center einen Rückruf des Nutzers zu initiieren.

### 4.2.1.4 Transaction

Zur bereits skizzierten Überwachung des Ablaufs einer Transaktion realisiert `Transaction` einen endlichen Automaten, dessen mögliche Zustände (`State`) und Zustandsübergänge (`Transition`) vom jeweiligen `Session`-Objekt aus der `TransactionTable` ausgelesen werden.

Die Initialisierung des Automaten findet statt, wenn einer durch den Nutzer angeforderten URL in der `SemanticsTable` eine Transaktions-ID zugeordnet ist, zu der noch keine `Transaction` an die `Session` gebunden ist (vgl. Abbildung 13).

Eine Transaktion wird immer durch mindestens zwei Zustände gebildet: einen Startzustand und einen Endzustand. Prinzipiell kann es mehrere Start- und Endzustände geben, insbesondere existieren i. Allg. weitere Zustände auf einem möglichen Pfad zwischen Start- und Endzustand.

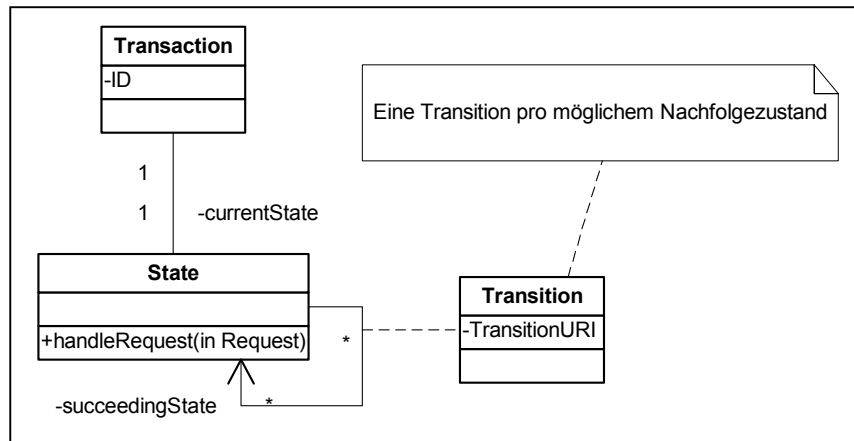
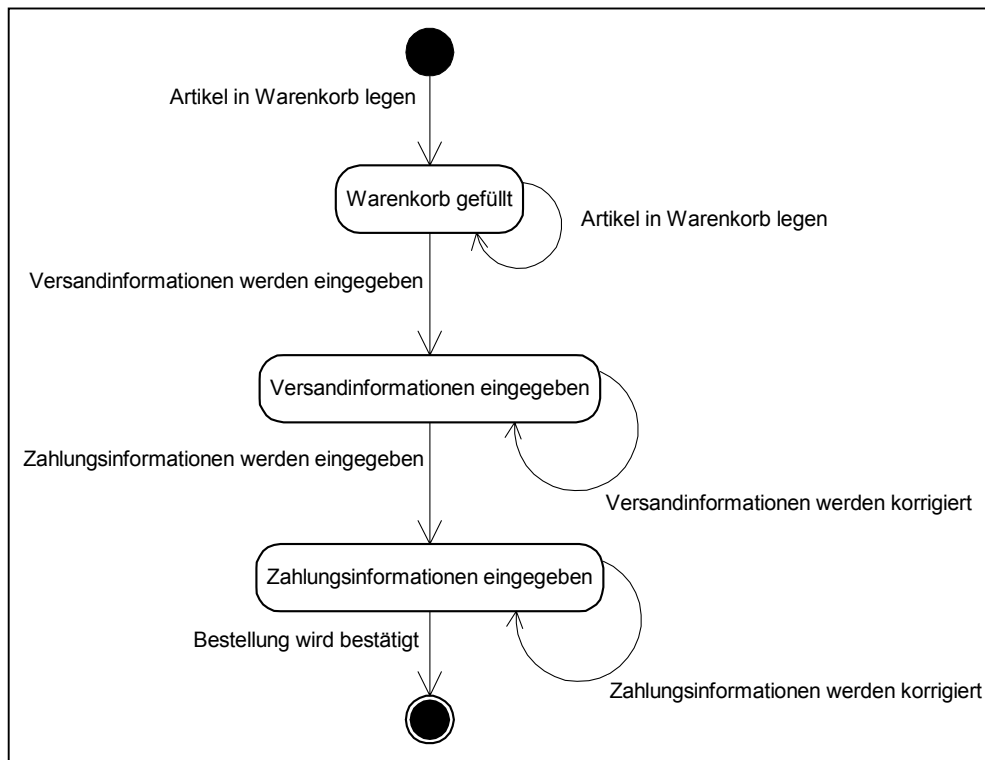


Abbildung 15: Klassendiagramm Transaction

Der statische Zusammenhang dieser Entitäten in Abbildung 15 macht deutlich, dass sich die Transaktion stets in genau einem Zustand befindet und Objekte vom Typ `Transition` die Zustandsübergänge zu allen möglichen Folgezuständen repräsentieren. Die Auswahl des Nachfolgezustands geschieht durch die Übergabe des Requests an den aktuellen Zustand, dessen Methode `handleRequest()` unter allen in Frage kommenden Transitionen diejenige ermittelt, die gemäß der angeforderten URL den eindeutig bestimmbar Nachfolgezustand herbeiführt.

Ein konkreter Automat zur Überwachung der in Abschnitt 4.1.1 skizzierten Online-Bestellung (3.a) ist in Abbildung 16 dargestellt.



**Abbildung 16: Zustandsdiagramm Online-Bestellung**

Es ist zu beachten, dass ein derartiger Automat den Workflow einer Transaktion für einen Nutzer nicht eins zu eins vorgibt. Im obigen Beispiel ist es denkbar, dass sich der Nutzer nach dem Eingeben der Versandinformationen weitere Artikel oder nicht zur Transaktion gehörende Webseiten anschaut, diese Requests werden jedoch nicht vom Transaktionsautomaten überwacht, sondern im Rahmen des einfachen Trackings durch die Session protokolliert und interpretiert. Der Automat übernimmt die Rolle eines Wächters, der genau dann aktiv wird, wenn ein Request ein Voranschreiten der Transaktion bedeutet, was durch die Geschäftslogik der Webanwendung exakt vorgegeben ist. So ist im obigen Beispiel das Hinzufügen weiterer Artikel zum Warenkorb nach der Eingabe der Versandinformationen nicht vorgesehen, obwohl dies grundsätzlich vorstellbar wäre. Der Automat muss deshalb so realisiert werden, dass er derartiges Fehlverhalten eines Nutzers in gleicher Weise behandelt, wie die zugrunde liegende Webanwendung, die beispielsweise einen Warnhinweis ausgibt und das Fortführen der begonnenen Transaktion vorschlägt.

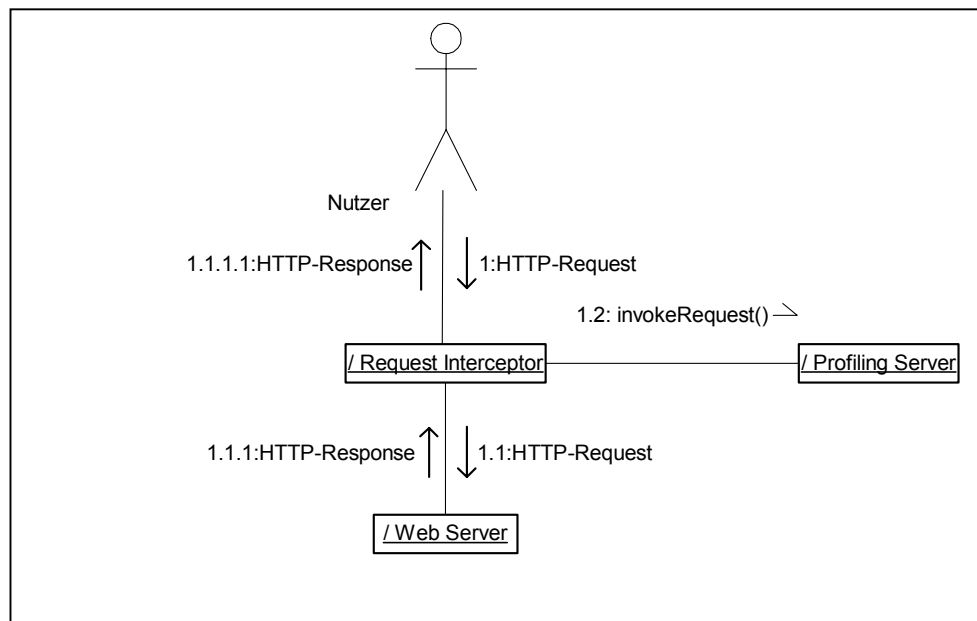
Durch diese Überwachungsfunktion des Transaktionsautomaten, die an die Requests geknüpft ist, die einen Zustandsübergang hervorrufen, ist der Startzustand einer Transaktion von untergeordneter Bedeutung. Im obigen Beispiel ist nur bekannt, dass der Nutzer zu Beginn der Transaktion noch keine Artikel in seinem Warenkorb hatte und über einen Link 'Zum Warenkorb hinzufügen' auf einer Produktseite den ersten relevanten Zustand der Transaktion erreicht hat. Mit der Initialisierung eines Transaktionsautomaten ist deshalb immer auch der erste Zustandsübergang verbunden.

## 4.2.2 Interaktionen zwischen Objekten

Nachfolgend werden zentrale Abläufe des Profiling und die dadurch bedingten Interaktionen zwischen den verschiedenen Objekten des Referenzmodells beschrieben, um neben der in Kapitel 4.2.1 dargelegten statischen Struktur die Funktionsweise eines darauf basierenden Systems zu verdeutlichen.

### 4.2.2.1 Request-Abspaltung und -Verarbeitung

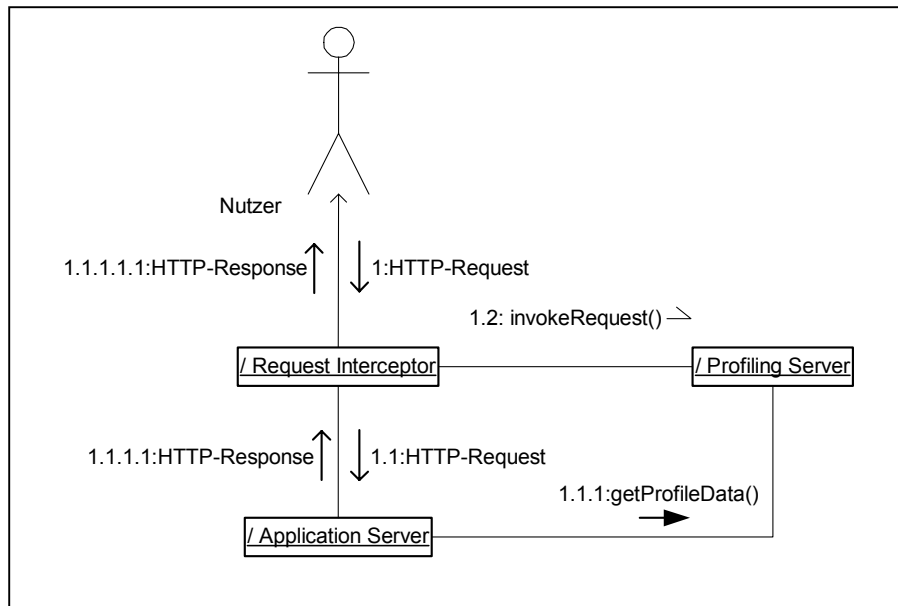
Abbildung 17 verdeutlicht die Abspaltung eintreffender Client-Requests des Nutzers am Request Interceptor als Kollaborationsdiagramm in Anlehnung an Abbildung 12, wobei die Verteilungsknoten als Rollen dargestellt sind.



**Abbildung 17: Kollaborationsdiagramm Request-Abspaltung**

Eintreffende HTTP-Requests werden dabei sowohl an den Webserver als auch asynchron an den Profiling Server weitergeleitet, d. h. es findet weder ein Puffern des Aufrufs noch ein Warten auf Aktivitäten des Profiling Servers statt. Lediglich die vom Webserver bereitgestellte Antwort auf den Request wird als HTTP-Response an den Nutzer zurückgeschickt. Damit entspricht diese Situation einem einfachen Webserver, zu dessen Nutzern individuelle Profile generiert werden.

Soll dagegen das Profilingssystem als Basis einer Personalisierung dienen, so ist ein weiterer Datenaustausch mit dem Profiling Server zum Zugriff auf die Profildaten notwendig, wie dies in Abbildung 18 dargestellt ist.



**Abbildung 18: Kollaborationsdiagramm Request-Abspaltung mit Personalisierung**

Eintreffende HTTP-Requests werden dabei wie zuvor sowohl an das Profilingssystem als auch an den für die Response-Erzeugung zuständigen Server weitergeleitet, hier ein Applikationsserver, der jedoch zur personalisierten Generierung einer Antwortseite zunächst Profilinginformationen erfragt, ehe ein Response erzeugt werden kann.

Die dargestellte Funktion `getProfileData()` steht dabei stellvertretend für die Funktionen der Schnittstellen `SessionIO` und `ProfileIO` (vgl. Abbildung 13), die vom Profilingssystem zum Zugriff auf Session- und Profildaten angeboten werden. Der Aufruf einer solchen Funktion macht in jedem Fall das Abwarten des Funktionsergebnisses erforderlich, dieses muss jedoch nicht zwangsläufig von einer erfolgten Verarbeitung des aktuellen Requests im Profilingssystem abhängen. Im Idealfall ist somit eine vollständig parallele Verarbeitung der Nutzeranfrage in beiden Systemen möglich; falls zur Personalisierung die Auswertung des aktuellen Requests durch den Profiling Server notwendig ist, muss an dieser Stelle besonderer Wert auf eine performante Implementierung des ggf. notwendigen Synchronisationsmechanismus gelegt werden, um die Antwortzeit des Gesamtsystems gegenüber dem Nutzer nicht unnötig lang werden zu lassen.

#### 4.2.2.2 Start einer Session

Beim Start des Web- bzw. Applikationsservers ist dafür zu sorgen, dass auch der Request Interceptor und das Profilingssystem in einen betriebsfähigen Zustand gebracht werden. Der Request Interceptor muss eingehende HTTP-Requests abspalten können, die an der `RequestImport`-Schnittstelle an das Profilingsystems übergeben werden. Dazu ist die initiale Erzeugung einer Instanz der Singleton-Klasse `SessionManager` notwendig.

Der Ablauf der danach möglichen Verarbeitung eintreffender Requests ist als Sequenzdiagramm in Abbildung 19 dargestellt:

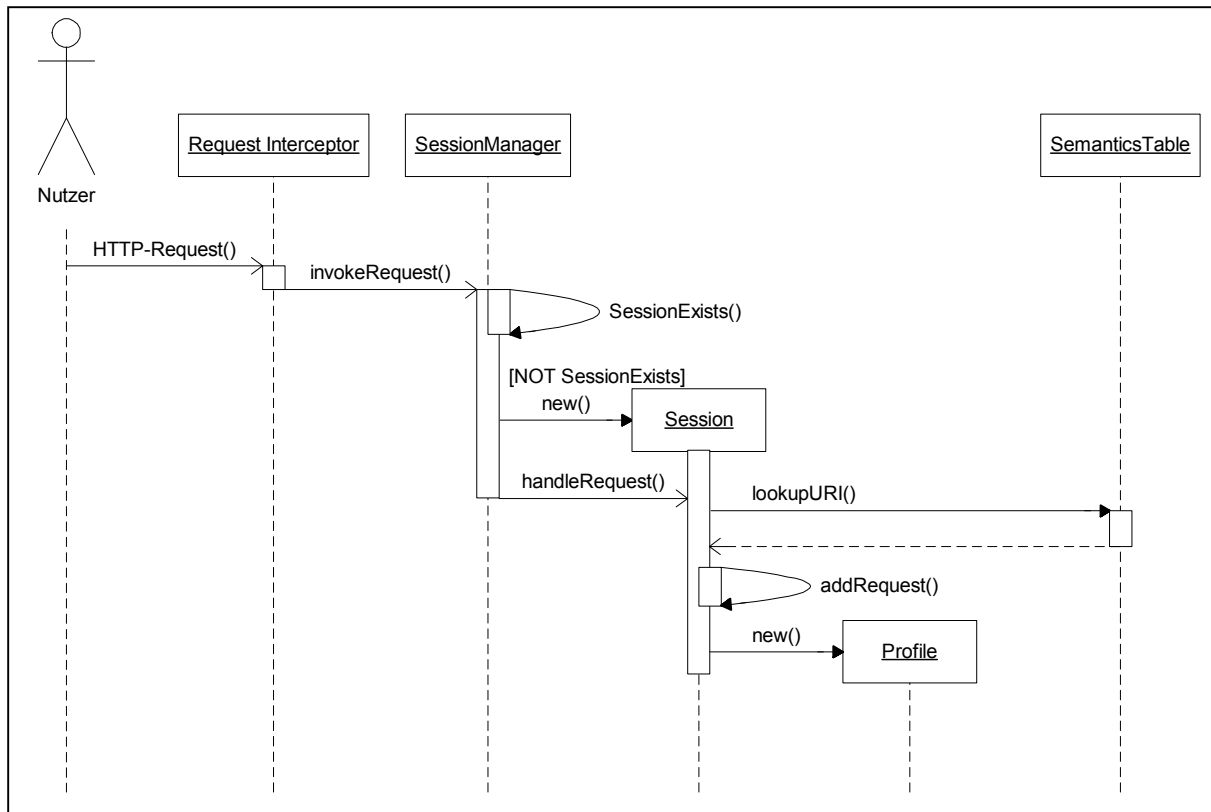


Abbildung 19: Sequenzdiagramm Sessionstart

- Der vom Request Interceptor abgespaltene Request wird mittels `invokeRequest()` dem `SessionManager` übergeben.
- Der `SessionManager` prüft, ob zu der übermittelten Session-ID bereits ein `Session`-Objekt existiert. Falls nicht, so ist die Session neu oder nach Überschreiten eines global einzustellenden Zeitlimits abgelaufen. In beiden Fällen wird ein neues `Session`-Objekt erzeugt, das fortan mit der übermittelten Session-ID referenziert werden kann. Eine mögliche Implementierung dieses Mechanismus besteht darin, zunächst zu versuchen, auf die Session zuzugreifen und das Erzeugen eines neuen `Session`-Objekts als Ausnahmebehandlung bei Nichtexistenz vorzusehen.
- Es erfolgt die eigentliche Requestverarbeitung, wie sie nachfolgend auch für jeden weiteren Request der Session abläuft. Anhand der abgerufenen URL klassifiziert `Session` den Request mittels `SemanticsTable` und fügt ihn in die eigene Datenstruktur zur Sammlung aller Requests einer Session ein.
- Falls in `SemanticsTable` die Anweisung `createProfile` hinterlegt war, wird ein neues Profil angelegt. Konnte anhand des ersten Requests bereits die ProfileID (PID) des Nutzers ermittelt werden, was beim Einsatz der in Kapitel 3.1.2 vorgestellten Cookies in vielen Fällen möglich ist, so kann dieses mittels `loadProfile` geladen werden. Alternativ dazu sollte die automatische Erzeugung eines anonymen Profils zu Beginn jeder Session voreingestellt werden können.

Falls die Authentifizierung eines bekannten Nutzers erst zu einem späteren Zeitpunkt der Session erfolgt, beispielsweise per Login, so ist das vorhandene Profil des Nutzers aus dem Persistenzspeicher zu laden und die bis dahin gewonnenen Informationen des anonymen Profils in dieses zu übertragen. Dies kann mithilfe der in Kapitel 2.3.2 beschriebenen Funktion 'Merge' durchgeführt werden, die zwei Profile zu einem aggregierten Profil verschmilzt und welche vom `ProfileManager` realisiert wird.

### 4.2.2.3 Profilerweiterung während einer Session

Zur Veranschaulichung der Abläufe bei einem einfachen Click eines Nutzers während seiner Session betrachten wir das Szenario 1.a aus Kapitel 4.1.1: ein Nutzer ruft eine Seite mit einer aktuellen Nachrichtenmeldung ab.

- Zunächst erfolgt die kaskadierende Weitergabe des Requests vom Request Interceptor über den RequestManager zum Session-Objekt, wobei Session und Profile in diesem Fall bereits in der Laufzeitumgebung des Systems existieren.
- Per `lookupURI()` der `SemanticsTable` kann der abgerufenen Seite zugeordnet werden, dass es sich um eine aktuelle Nachrichtenmeldung handelt und zu welchem Themengebiet der Inhalt gehört.
- Diese Informationen werden `Session` zugefügt.

Nun bestehen zwei Möglichkeiten, das Interesse des Nutzers an Nachrichten und dem betreffenden Thema in das ihn repräsentierende Profil zu schreiben und es damit zu erweitern:

1. Der Profileintrag geschieht unmittelbar nach der Zuordnung einer Bedeutung. Dieser Ablauf ist in Abbildung 20 dargestellt.
2. Die Profilerweiterung geschieht erst am Ende der Session, wobei dann die gewonnenen Informationen aller Requests gesammelt in das Profil geschrieben werden.

Der Vorteil der erstgenannten Möglichkeit ist, dass das korrespondierende Profil eines Nutzers zu jedem Zeitpunkt den gesamten Wissensstand über ihn repräsentiert, welcher dann über die Schnittstelle `ProfileIO` zugreifbar ist.

Die zweite Alternative erfordert dagegen ein Auslesen der aktuellsten Interaktionen aus der `Session` des Nutzers über die Methoden der Schnittstelle `SessionIO`; diese Trennung des protokollierten Verhaltens und der damit verbundenen Präferenzen ist jedoch ohnehin sinnvoll, da Besuche eines einzelnen Nutzers der gleichen Website aufgrund unterschiedlicher Anliegen sehr verschieden verlaufen können und die auf Profiling basierenden Maßnahmen sich primär am Verhalten der aktuellen Sitzung orientieren sollten.



Je nach Art der Implementierung kann das gebündelte Erweitern von Profilen auch aus Laufzeitgründen geboten sein.

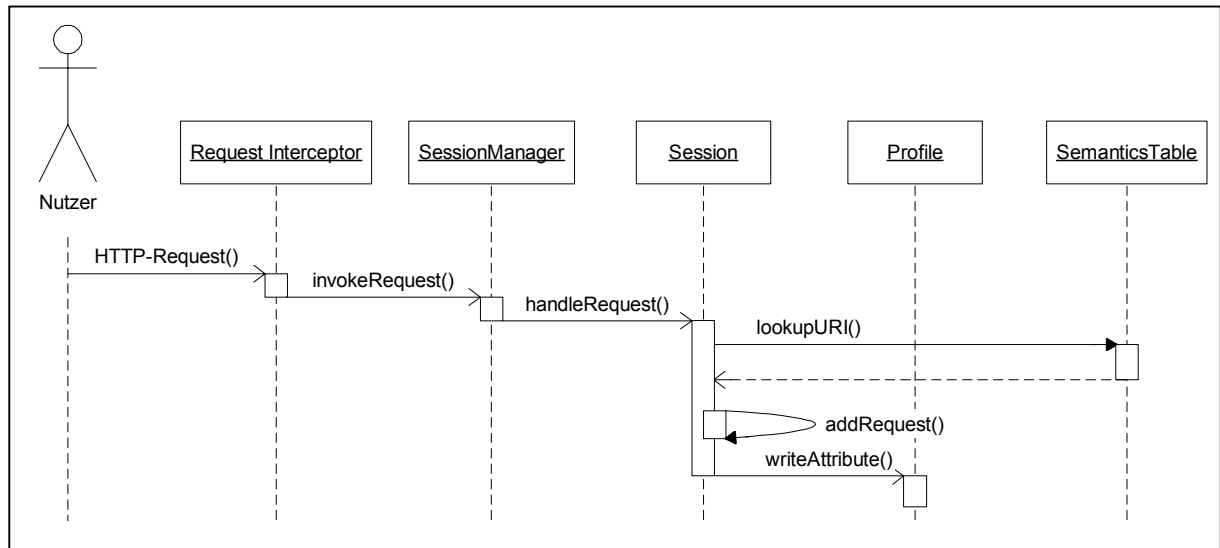


Abbildung 20: Sequenzdiagramm Profilerweiterung

#### 4.2.2.4 Durchführung einer Transaktion

Neben einfachen Seitenabrufen ist vor allem der Ablauf von Transaktionen für das Profiling von Interesse. Aus Sicht der daran beteiligten Objekte kann die Transaktionskontrolle wie in Abbildung 21 skizziert beschrieben werden.

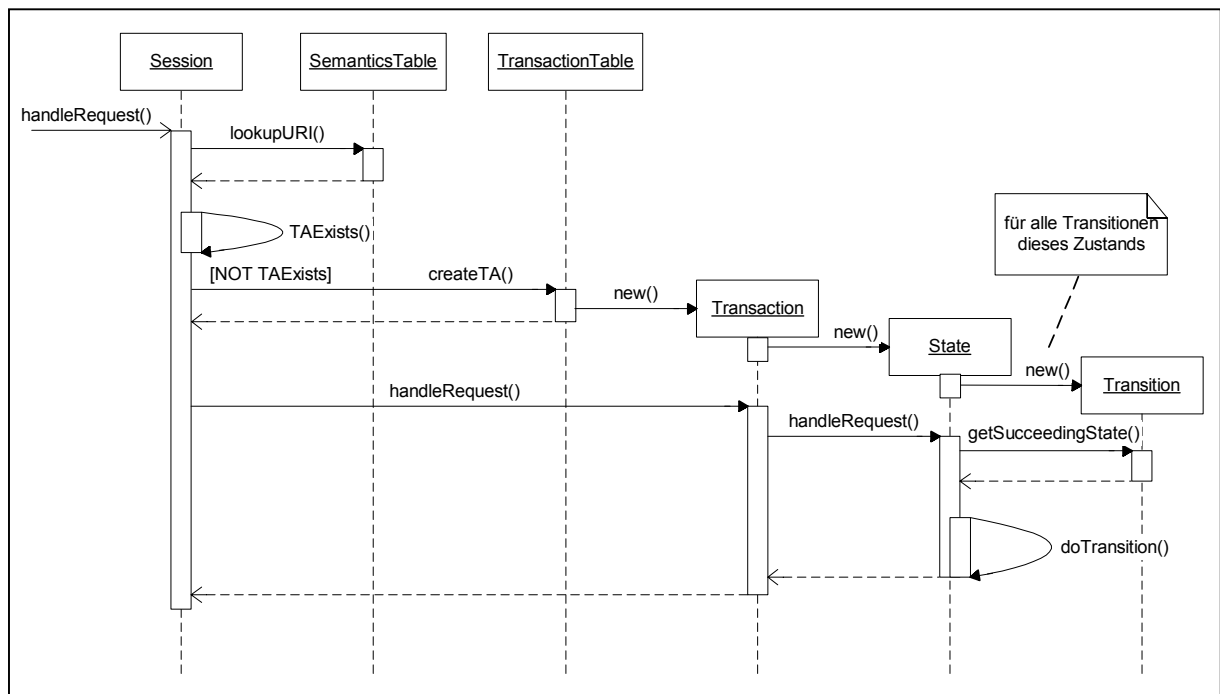


Abbildung 21: Sequenzdiagramm Transaktion

- Der Aufruf der Methode `lookupURI()` der `SemanticsTable` liefert anstelle oder zusätzlich zur semantischen Klassifikation die Kennung eines Transaktionstyps zurück.

- Das `Session`-Objekt prüft, ob bereits ein `Transaction`-Objekt dieser Art für den Nutzer existiert, falls nicht, wird ein diesen Transaktionstyp abbildendes Objekt mittels `TransactionTable` erzeugt. Dabei werden kaskadierend die zum Transaktionsautomaten gehörenden Zustände instanziiert, welche wiederum die von ihnen ausgehenden Transitionen als Objekte erzeugen.
- Anschließend folgt wie für alle weiteren Requests zu diesem Transaktionstyp die direkte Weitergabe des `Request`-Objekts an `Transaction`, das den in Kapitel 4.2.1.4 skizzierten Überwachungsautomat der Transaktion realisiert. `Transaction` besitzt mit dem Attribut `currentState` (vgl. Abbildung 15) eine Referenz auf seinen aktuellen Zustand, welcher anhand der URL des Requests die durchzuführende Transition bestimmen kann und deren Methode `getSucceedingState()` den korrekten Nachfolgezustand von `Transaction` zurückliefert.
- Schließlich sorgt `State` dafür, in den Nachfolgezustand zu konvertieren und den erfolgreichen Zustandsübergang an das `Transaction`-Objekt zurückzumelden.

An dieser Stelle ist eine Fehlerbehandlung vorzusehen, falls eine Transition nicht wie vorgesehen ablaufen konnte, beispielsweise weil kein passendes `Transition`-Objekt und damit kein gültiger Nachfolgezustand gefunden werden konnte, was u. a. aus irregulärem Nutzerverhalten folgen kann.

Die Zuweisung von Semantikschlüsseln oder Anweisungen an die URL eines solchen Requests bleibt vom Transaktionsablauf unberührt. Somit ist die Durchführung einer vom Anwender des Profilingssystems frei definierbaren Aktion möglich, insbesondere ein Profileintrag, da ein einzelner Schritt einer Transaktion meist mit der Gewinnung für das Profiling wichtiger Informationen einhergeht.

Bezogen auf das Beispiel 3.a aus Kapitel 4.1.1 ist aus Marketingsicht auch von Interesse, wenn Nutzer einen Artikel in den virtuellen Warenkorb legen, der Einkaufsvorgang dann jedoch abgebrochen wird oder der Artikel vor der Bestellung aus dem Warenkorb entfernt wird. Die kurzfristige Erwägung eines Kaufs lässt auf ein vorhandenes Interesse an dem Artikel schließen. Das Protokollieren derartigen Verhaltens und die separate Speicherung der nicht beendeten Transaktion kann beispielsweise genutzt werden, um bei einer späteren Preissenkung dieses Artikels individuelle verkaufsfördernde Maßnahmen zu ergreifen.

### 4.2.2.5 Ende einer Session

Das explizite Ende der Sitzung eines Nutzers wird durch einen Logout-Vorgang erkannt. Ein derartiger Ablauf ist in Abbildung 22 dargestellt.

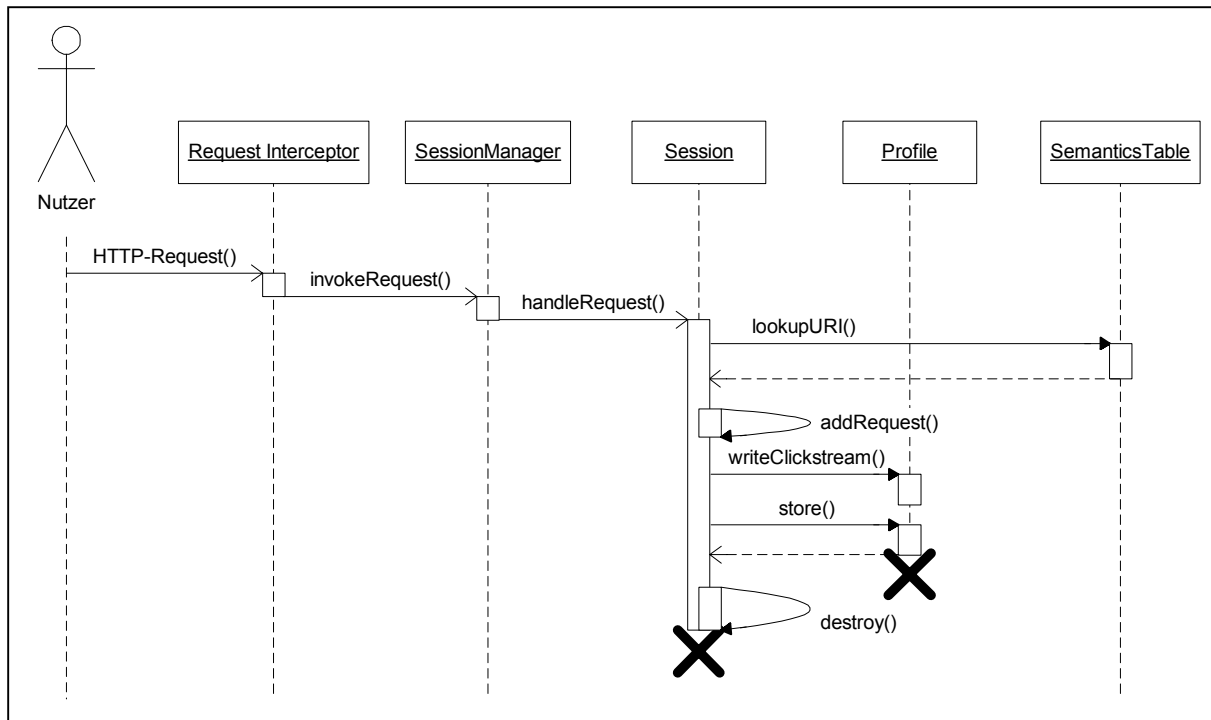


Abbildung 22: Sequenzdiagramm Sessionende

- Das Betätigen eines Logout-Buttons durch den Nutzer wird durch Nachschlagen in der SemanticsTable erkannt und führt neben der normalen Registrierung des Clicks in Session mittels `addRequest()` zur Durchführung der durch `stopSession` definierten Verarbeitungsschritte.
- Die Methode `writeClickstream()` führt zur Übertragung aller Request-Daten dieser Session zur Erweiterung von Profile. Falls die mit den einzelnen Request-Objekten verbundenen Semantics nicht bereits während der Session in das Profil des Nutzers geschrieben wurden, muss dies nun ebenfalls geschehen.
- Der Aufruf von `store()` sorgt für die persistente Speicherung des gesamten Nutzerdatensatzes, woraufhin die entsprechenden Instanzen von Profile und Session in der Laufzeitumgebung des Systems zerstört werden können.

Aufgrund der Eigenschaften des HTTP-Protokolls kann nicht sichergestellt werden, dass Nutzer einen vorgesehenen Logout-Vorgang durchführen. Deshalb empfiehlt sich die zusätzliche Realisierung eines in der Praxis von Webanwendungen üblichen Timeout-Mechanismus: nach dem Ablauf einer global voreingestellten Zeitspanne, in der zu einem Nutzer kein Request auftrat, werden die o. g. abschließenden Verarbeitungsschritte automatisch veranlasst.

#### 4.2.3 Zusätzliche Komponenten

In diesem Kapitel werden abschließend die Grundideen verschiedener Komponenten vorgestellt, welche nicht primär zu einem System für Profiling gehören, dieses jedoch sinnvoll ergänzen bzw. dessen Einsatz erst praktisch handhabbar machen.

Die ersten vier der hier skizzierten Werkzeuge bieten grundlegende Funktionalitäten für den Einsatz des bisher entworfenen Modells für ein Profilingssystem und arbeiten größtenteils auf

dessen Modellen und Daten, weshalb sich eine zweischichtige Implementierung anbietet: ein Client-Werkzeug bildet die Nutzerschnittstelle für den Business Manager, der damit eine einfach zu bedienende grafische Oberfläche zur Manipulation des Profilschemas und der verwendeten Kategorisierungen, zur Semantikdefinition und zur Analyse und Visualisierung von Profildaten erhält; das Gegenstück der einzelnen Werkzeuge bilden Softwarekomponenten im Profiling Server, welche dedizierte Schnittstellen für die verschiedenen Client-Komponenten anbieten und die gewünschten Operationen serverseitig ausführen.

Die Client-Werkzeuge kommunizieren per Netzwerk mit dem Profiling Server und sind softwaretechnisch zu einem 'Profiling Administration Client' zusammengefasst (vgl. Abbildung 12).

### 4.2.3.1 Schema Editor

Der Schema Editor dient zum Editieren und Verwalten des im Profilingssystem zu verwendenden Profilschemas und der semantischen Klassifikationen. Profilschema und Klassifikationen basieren dabei auf dem Entwurfmuster 'Kompositum' aus [GHJV96], dessen Verwendung für `Classification` bereits in Kapitel 4.2.1.3 beschrieben wurde.

Abbildung 23 macht deutlich, dass sowohl ein Profilschema als auch daraus abgeleitete Profilinginstanzen ebenfalls durch Objekte des Kompositum-Musters geeignet repräsentiert werden können, wenn ähnlich der Datenstruktur 'Baum' Nutzdaten in jeden Knoten der Komposition geschrieben werden.

Der Schema Editor erlaubt nun das Anlegen und Editieren eines Profilschemas und damit die Spezifikation aller Daten, die zu einem Nutzer individuell gespeichert werden sollen. Auf diese Weise kann die bereits in Abbildung 11 in Form eines Klassendiagramms vorgeschlagene Struktur für ein Profilschema in eine Hierarchie von Klasseninstanzen von `Composition` und `Leaf` überführt werden.

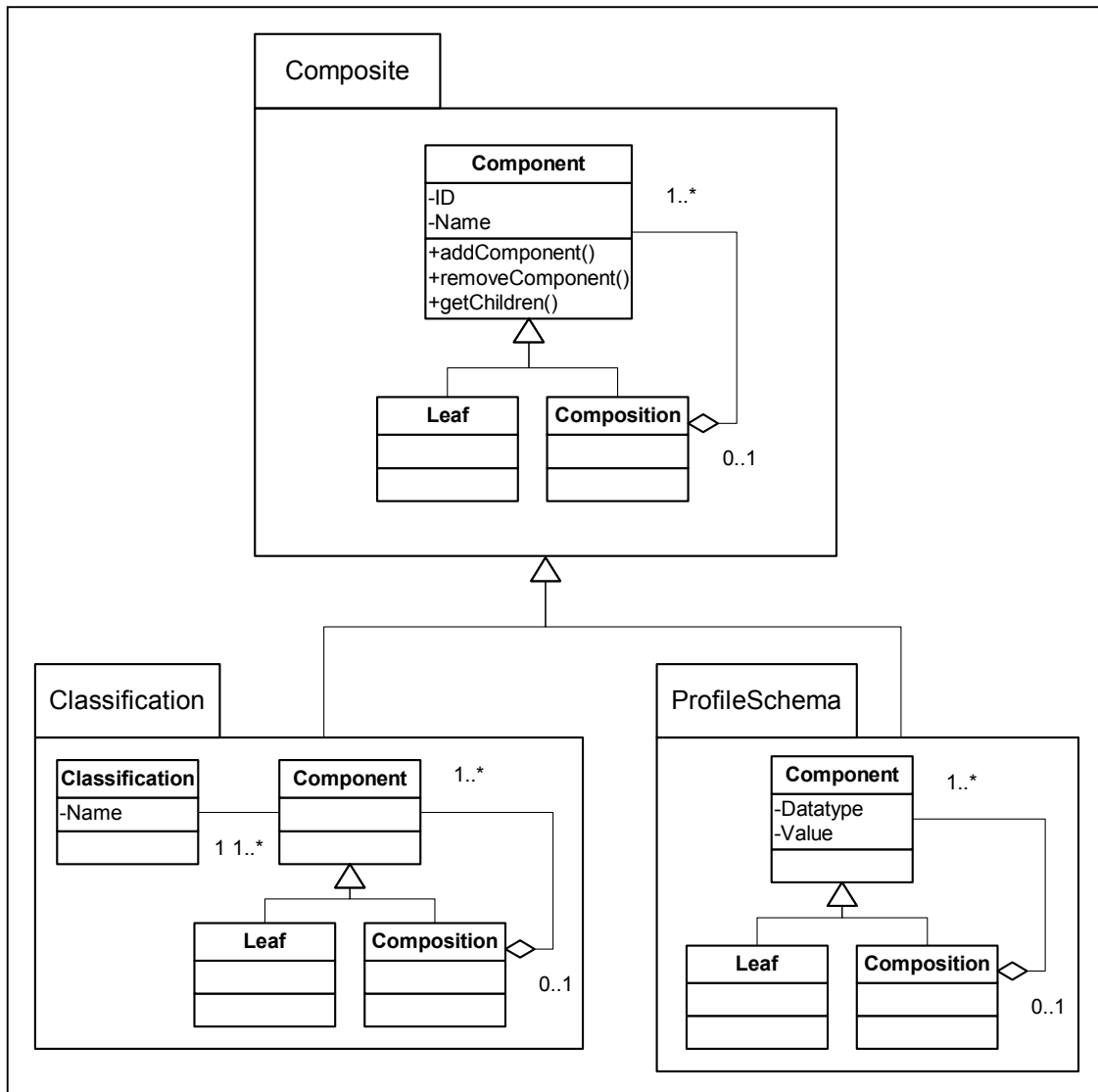


Abbildung 23: Klassendiagramm Composite

Diese Repräsentation eines Profilschemas kann persistent gemacht werden und dient als inhaltsleerer Prototyp zur Erzeugung und Erweiterung darauf basierender Profile. Dazu wird beim Neuanlegen eines Profils eine leere Kopie von `ProfileSchema` angelegt, deren `Component`-Objekte gemäß der im `ProfileSchema` vordefinierten Datentypen mit konkreten Werten belegt werden können. Eine beispielhafte Komposition von Klasseninstanzen, welche das in Abbildung 11 vorgeschlagene Profilschema berücksichtigt, ist in Abbildung 24 dargestellt.

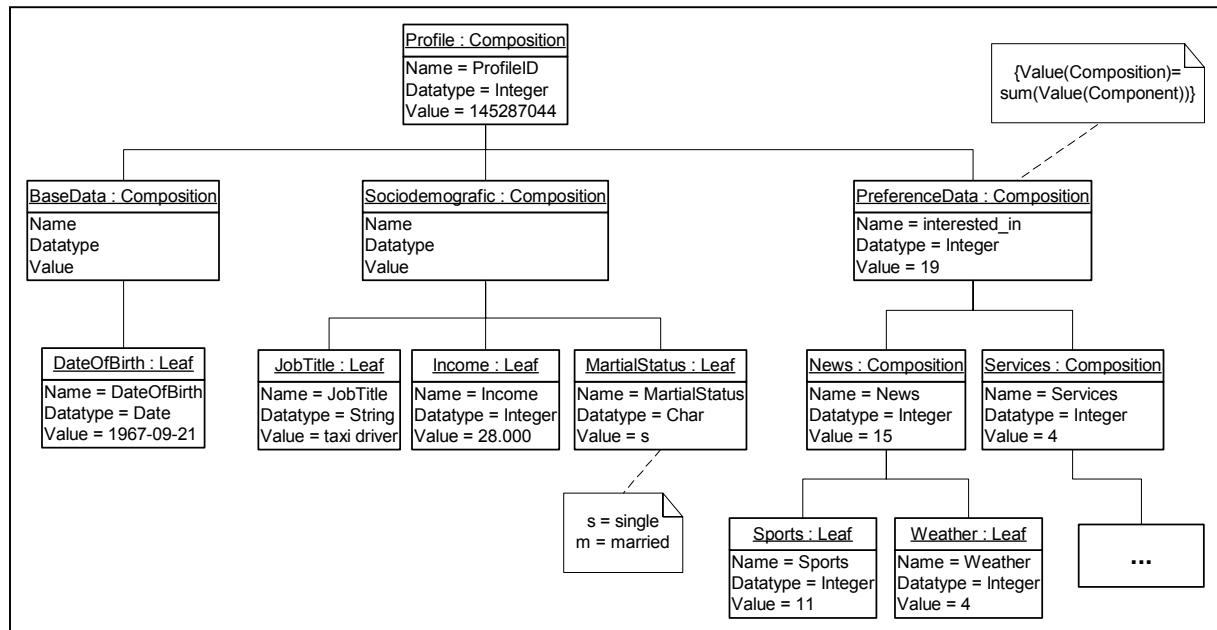


Abbildung 24: Klassendiagramm Profilinstanz

Es wird vorgeschlagen, nur einen Teil der im Profilschema vorgesehenen Attribute beim Neuanlegen eines Profils als Objekte zu erzeugen, um Speicherplatz sowohl in der Laufzeitumgebung als auch im Persistenzspeicher zu sparen. Werden im Rahmen von Nutzerinteraktionen neue Daten gewonnen, so kann das Profil gemäß der Struktur des Profilschemas um die entsprechenden Attribute erweitert werden.

Mit dem gleichen Werkzeug können analog auch eine oder mehrere zu verwendende semantische Klassifikationen angelegt und editiert werden, sofern keine Standardklassifikation wie DDC oder UDC importiert wird.

Eine geeignete Klassifikation für die Website eines Online-Shops kann beispielsweise aus den Warengruppen und Artikeln bestehen, die offeriert werden. Auf diese Weise lassen sich alle Webseiten, auf denen Warengruppen oder Artikel dargestellt werden, leicht klassifizieren. Seitenabrufe eines Nutzers werden dann in einer der Klassifikation gleichenden Struktur im Bereich *PreferenceData* (vgl. Abbildung 11) gezählt.

Eine weitere Möglichkeit zur Bildung einer passenden Klassifikation besteht darin, die auf einer Website vorkommenden Themen strukturiert abzubilden, wozu ggf. die Sitestruktur adaptiert werden kann. Dieses Vorgehen eignet sich jedoch nur für Websites, deren Content thematisch fixiert ist, da die im Profilingssystem verwendete Klassifikation möglichst selten geändert und im günstigsten Fall ausschließlich erweitert werden sollte.

In Abbildung 24 wird auch deutlich, wie Derivative einer Klassifikation dazu verwendet werden können, um Präferenzdaten eines Nutzers zu repräsentieren, indem ihre hierarchische Struktur in das Profilschema eingefügt wird. In diesem Beispiel leitet sich die Klassifikation von den auf der Website gebotenen Inhalten ab, u. a. Nachrichten (News) und weitere Leistungen (Services). Das hier beschriebene Profil lässt erkennen, dass der Interessensschwerpunkt des Nutzers im Bereich der Nachrichten liegt und hier besonders bei den Sportmeldungen. Die verwendeten Ganzzahlwerte können einzelne Seitenabrufe beziffern oder sich aus verschiedenen anderen Kriterien ergeben, beispielsweise einer gewichteten Kombination aus Seitenabruf, Verweildauer und sonstigen Interaktionen. Diese Gewichtung kann im Zuge der Klassifizierung einer Seite festgelegt und in *SemanticsTable* gespeichert werden.

Das hier vorgestellte Konstrukt der hierarchischen Profilschemata und Klassifizierungen ermöglicht in flexibler Weise die Erfüllung der in Kapitel 2.2.2 aufgestellten Anforderungen an die Profildatenhaltung hinsichtlich Erweiterbarkeit, Strukturierung und Werteskalierung.

Die Existenz derartiger Objektstrukturen zur Laufzeit des Profilingssystems, welche das Profilschema und die verwendeten Klassifizierungen repräsentieren, kann nicht nur zu deren Manipulation und Erstellung darauf basierender Personalisierungsregeln genutzt werden, sondern ermöglicht auch deren Umwandlung in andere Formate, um den in Kapitel 2.2.3 genannten Erfordernissen des Datenaustauschs gerecht zu werden.

Als flexibles Austauschformat zur maschinenlesbaren Beschreibung derartiger Strukturen eignet sich XML, darüber hinaus ist eine Automatisierung der Erzeugung der erforderlichen Datenbankstrukturen wünschenswert. Schließlich gilt es auch, aus in XML oder in Datenbanken gespeicherten Klassifizierungen und Profilschemata die Objektstruktur zur Laufzeit aufzubauen.

Daraus ergeben sich nachfolgend genannte Anforderungen, die [GHJV96] als Indikatoren für die Anwendung des Entwurfsmusters 'Builder' propagiert:

- "Der Algorithmus zum Erzeugen eines komplexen Objekts soll unabhängig von den Teilen sein, aus denen das Objekt besteht und wie sie zusammengesetzt werden."
- "Der Konstruktionsprozeß muß verschiedene Repräsentationen des zu konstruierenden Objekts erlauben."

Eine mögliche Umsetzung dieses Musters für die hier gegebene Aufgabe ist in Abbildung 25 dargestellt.

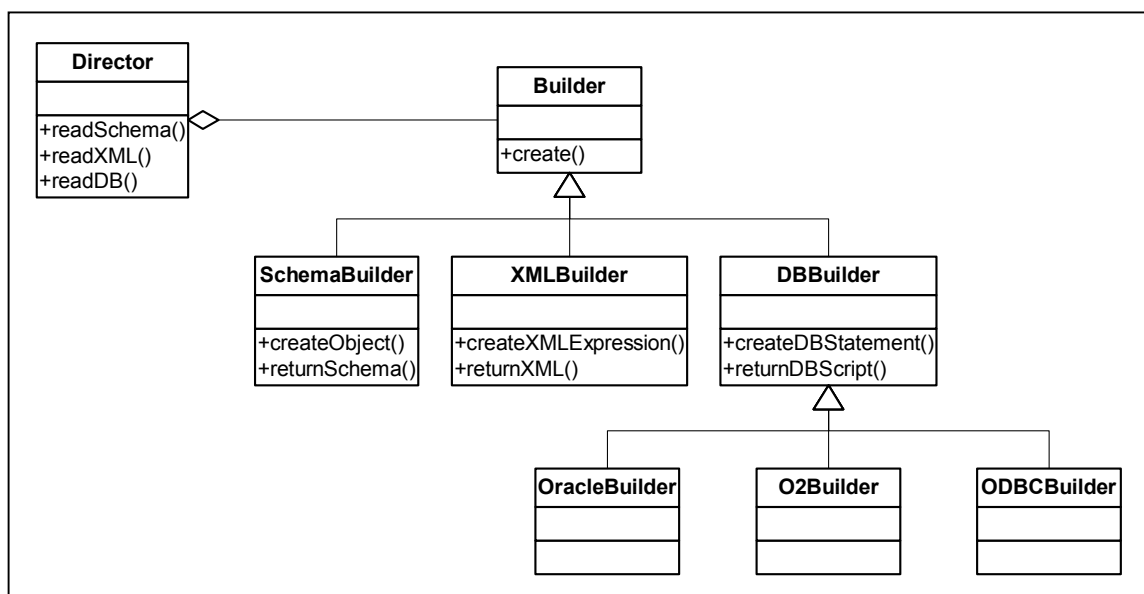


Abbildung 25: Klassendiagramm SchemaBuilder

Die Klasse `Director` verfügt, ggf. mittels Unterklassen, über die Möglichkeit, Schemata als Objektstrukturen zu explorieren, in XML-Dateien zu parsen und aus Datenbanken zu lesen. Die Serverkomponente des Schema Editor konfiguriert das Erzeugersystem mit den

gewünschten Zielformaten und veranlasst durch Aufruf der `read...()`-Methode von `Director` für das vorliegende Quellformat die schrittweise Konstruktion der Zielobjekte.

Auf diese Weise kann der Import und Export von Profilschemata und Kategorisierungen in Form von XML realisiert werden, wozu `Director` zum Durchlaufen der Objektstrukturen in Art einer Breitensuche insbesondere die Methode `getChildren()` von `Component` als Schnittstelle nutzt, welche aus Abbildung 23 ersichtlich wird.

Nach der initialen Erzeugung eines Profilschemas mithilfe der grafischen Oberfläche des Schema Editor liegt dieses als Struktur aus `Composition`- und `Leaf`-Objekten vor. Ein mögliches objektrelationales Mapping, um gefüllte Profile persistent speichern zu können, besteht in dem Vorgehen, für jedes `Composition`-Objekt eine Tabelle in einer relationalen Datenbank anzulegen, deren Spalten entweder die konkreten Werte der Kindobjekte vom Typ `Leaf` enthalten oder im Falle einer weiteren Komposition eine Referenz auf den Eintrag der diesem `Composition`-Objekt zugeordneten Tabelle.

Dieses Beispiel belegt die prinzipielle Realisierbarkeit des Entwurfs, wobei die Freiheit besteht, die persistente Datenhaltung den jeweiligen Gegebenheiten und Anforderungen entsprechend anders zu gestalten. So ist nach Abbildung 25 auch die Verwendung einer Objektdatenbank (O2) möglich, wobei die Flexibilität des Entwurfsmusters 'Builder' die Erzeugung einer Vielzahl datenbankspezifischer SQL-DDL-Anweisungen erlaubt.

### 4.2.3.2 URI Mapper

Der URI Mapper ist ein Client-Werkzeug zur Klassifizierung von Inhalten der Website. Er ermöglicht die Zuordnung eines oder mehrerer Elemente der semantischen Klassifikation zu einzelnen HTML-Seiten oder darin eingebetteten Elementen, anhand derer eine automatische Profilerweiterung bei Abruf durch einen Nutzer erfolgt.

Dazu muss eine Klassifikation vorhanden sein, die es einerseits erlaubt, möglichst viel des dargebotenen Contents bedeutsam einzuordnen und auf die andererseits innerhalb des verwendeten Profilschemas Bezug genommen werden kann. Zu diesem Zweck kann eine Klassifikation standardmäßig an der Stelle des Objekts `PreferenceData` des in Abbildung 11 vorgeschlagenen Profilschemas eingefügt werden, wie dies am Beispiel einer Profilinanz in Abbildung 24 geschehen ist. Die Bedeutung dieses Zweigs eines Profils wird im Knoten `PreferenceData` hinterlegt; i. Allg. bietet sich an, die Interessen eines Nutzers auszudrücken, indem die zugeordnete Semantik seiner Seitenabrufe gespeichert wird.

Die Serverkomponente des URI Mapper exploriert die Objektstruktur von `Classification`, welche innerhalb der grafischen Nutzeroberfläche der Client-Komponente beispielsweise als Verzeichnisbaum dargestellt wird. In einem weiteren Fenster greift der URI Mapper mit einem integrierten Browser per HTTP auf die Website zu, sodass ein redaktioneller Mitarbeiter pro abgerufener Seite lediglich einen Schlüssel der semantischen Klassifikation auswählen muss, der dann in `SemanticsTable` der URL zugeordnet wird.

Grundsätzlich ist die gewichtete Adjunktion mehrerer Schlüssel zu einer URL zu ermöglichen, sowie die Einbeziehung weiterer im Rahmen der Übermittlung der Client-Requests ermittelbaren Daten, z. B. die berechnete Verweildauer des Nutzers auf einer Seite. Aufgrund der Überlegungen in Kapitel 2.2.2 sollten rationale Skalen mit positivem oder



positivem und negativem Wertebereich zur Repräsentation von Nutzerpräferenzen verwendet werden. Auch der mehrfache Einsatz der gleichen oder verschiedener Klassifikationen ist bei der Realisierung eines auf diesem Modell basierenden Profilingssystems zu berücksichtigen.

Betrachtet man das Anwendungsbeispiel 1.b aus Kapitel 4.1.1, Abruf eines redaktionellen Artikels durch einen Nutzer, so könnte die Betrachtung der Zusammenfassung des Artikels mit der Erhöhung des das Nutzerinteresse repräsentierenden Wertes um 1,0 Indexpunkte einhergehen, während der Abruf der ausführlichen Fassung mit Verweildauer größer 60 Sekunden als besonders hohes Interesse interpretiert wird und deshalb ein Faktor von 2,5 zugeordnet wird.

Negative Werte können Abneigungen des Nutzers widerspiegeln, die durch Online-Befragungen oder Voting ermittelt werden. Eine äquivalente Art der Informationsspeicherung besteht darin, dieselbe Klassifizierung einmal mit der Bedeutung 'hat Interesse an' und einmal mit 'hat kein Interesse an' in das Profilschema einzufügen.

Besteht die Klassifizierung aus Warengruppen und Artikeln eines Online-Shops, so kann mit der mehrfachen Verwendung zwischen 'interessiert sich für' (s. Anwendungsszenario 1.c in Kapitel 4.1.1) und 'hat bereits gekauft' unterschieden werden, was für eine effektive Personalisierung von großer Bedeutung ist.

Eine Implementierungsalternative für das multiple Einbinden der gleichen Klassifikation in das Profilschema liegt in der Speicherung mehrerer Werte in jedem `Component`-Objekt.

Da technisch betrachtet zwischen einfachen Seitenabrufen und Transaktionen unterschieden werden muss, dient der URI Mapper auch zur Bekanntmachung der Abläufe möglicher Transaktionen gegenüber dem Profilingssystem. Dazu werden analog zur Zuordnung von Klassifikationsschlüsseln die einzelnen Schritte einer Transaktion durchlaufen und die korrespondierenden Zustände definiert, wobei sich mögliche Transitionen aus der Abfolge von Seitenabrufen ergeben, die zu einem Zustandwechsel führen. Derart beschriebene Transaktionen werden in `TransactionTable` gespeichert.

Sowohl bei einfachen Seitenabrufen als auch bei Transitionen innerhalb einer Transaktion soll die Ausführung zusätzlicher Aktionen neben der durch `SemanticsTable` definierten Profilerweiterung möglich sein. Dabei kann es sich um zusätzliche Profileinträge handeln, beispielsweise die Übernahme persönlicher Daten des Nutzers aus einem ausgefüllten HTML-Formular (Name, Anschrift, Bestelldaten usw.), aber auch ein Benachrichtigungsmechanismus ist wünschenswert, der z. B. automatische Datenexporte zu anderen Anwendungen oder systemübergreifende Notifikationen veranlasst. Hierzu können im URI Mapper ebenfalls die bereits beschriebenen Anweisungen vom Typ `Instruction` definiert werden.

Als anspruchsvolle Herausforderung müssen URI Mapper und `SemanticsTable` komplexe zusammengesetzte URLs verarbeiten können, wie sie bei der dynamischen Generierung von HTML-Seiten vorkommen. Dazu sind die in den Kapiteln 3.4.2 ff. skizzierten Vorgehensweisen zu realisieren, insbesondere ein Pattern-Matching-Algorithmus, der durch reguläre Ausdrücke spezifizierte Zeichenfolgen in einer URL erkennen, extrahieren und mithilfe externer Datenquellen auswerten kann. Hierzu sind ggf. Möglichkeiten zur Einführung zusätzlicher Hilfsdatenstrukturen vorzusehen, beispielsweise Tabellen, die

Artikelnummern eines Online-Shops auf Warengruppenkennungen abbilden, die zur Klassifizierung verwendet werden.

Die bereits in Kapitel 3.4.3 erwähnte beispielhafte URL einer Webseite zur Anzeige eines durch Warengruppen- und Artikelnummer spezifizierten Produkts eines Online-Shops kann damit folgendermaßen verarbeitet werden:

Das Nachschlagen der per HTTP-Request übermittelten Adresse

`www.p-buy.de/catalog.jhtml?action=showitem&cat_id=472&product_id=015926`

mit der Methode `lookupURI()` in `SemanticsTable` führt zum Matching mit dem regulären Ausdruck

`/catalog.jhtml?action=showitem&cat_id=\*=x1\&product_id=\*=x2\`

wobei durch die Bezeichner `\*=x1\` und `\*=x2\` die an den jeweiligen Stellen der realen URL vorgefundenen Zeichenketten zwei temporären Variablen zugewiesen werden. Diese können unmittelbar in das Nutzerprofil übernommen werden oder bedürfen ggf. einer weiteren Dereferenzierung. Im ersten Fall ist in `SemanticsTable` beispielsweise hinterlegt, dass ein direkter Profileintrag erfolgen kann, da die Warengruppen- und Artikelnummern für die semantische Klassifizierung verwendet werden:

**SemanticsTable**

URI	Semantics (S,I,T)
<code>/catalog.jhtml?action=showitem&amp;cat_id=\*=x1\&amp; product_id=\*=x2\</code>	<code>S=Profile.interested_in. products.category(x1).product(x2)</code>

Gemäß der Modellierung in Abschnitt 4.2.1.3 ist neben der Zuweisung semantischer Schlüssel (`Semantics`) zu einer URL auch die Zuweisung von Anweisungen (`Instruction`) und Transaktionen (`Transaction`) erlaubt, was hier durch die Buchstaben `S`, `I` und `T` ausgedrückt wird. Falls die temporären Variablen weiter aufgelöst werden müssen, ist das Nachschlagen in zusätzlichen Tabellen zu ermöglichen:

**SemanticsTable**

URI	Semantics (S,I,T)
<code>/catalog.jhtml?action=showitem&amp;cat_id=\*=x1\&amp; product_id=\*=x2\</code>	<code>S=Profile.interested_in. DDC(TableLookup('Category',x1)) S=Profile.interested_in. DDC(TableLookup('Product',x2))</code>

In diesem Fall führt das Interesse des Nutzers zu zwei gleichrangigen Einträgen im Präferenzbereich seines Profils, wobei die jeweiligen Schlüssel der in Abschnitt 3.4.1.2 vorgestellten Dewey Decimal Classification (DDC) verwendet werden, die der Warengruppe und dem Produkt in einer Hilfstabelle zugeordnet sind, für `Category` also beispielsweise:

**Category**

Variable	S (DDC)
...	...
471	053
472 ----->	512
473	513
...	...

Der grundlegende Nachteil dieses Referenzmodells, dass die Applikationslogik der Webanwendung im Profilingssystem teilweise nachgebildet werden muss, lässt sich durch geeignete Import- und Zuordnungsmechanismen ausgleichen, die im URI Mapper realisiert werden.

Primäres Ziel bei der Anwendung eines solchen Systems muss es jedoch sein, insbesondere die Klassifizierung von Content durch den Einsatz von CMS in Verbindung mit Keywords, RDF und ähnlichen Mitteln effektiv zu automatisieren, welche bereits in Kapitel 3.4 beschrieben wurden.

#### 4.2.3.3 Analyzer

Als weitere zusätzliche Komponente empfiehlt sich die Implementierung eines Analysewerkzeugs, welches auf Basis der im Profilingssystem gespeicherten Informationen und den zur Verfügung gestellten Schnittstellen die in den Kapiteln 2.3.3 und 2.4 skizzierten Analysen und Visualisierungen von Profildaten realisiert.

Zur Exploration der potenziell zur Verfügung stehenden Informationen greift die Serverkomponente des Analyzer auf die Objektinstanz von `ProfileSchema` zu. Das Lesen großer Mengen von Profilen sollte über eine eigene Schnittstelle zur Profildatenbank erfolgen, da das Auslesen mittels Instanziierung einzelner Profile in der Laufzeitumgebung des Profiling Servers als wenig performant erwartet werden kann.

Es ist zu beachten, dass der Ablauf von Analysen einerseits zur Erstellung neuer, statistischer Profile führen kann (z. B. Durchschnittsnutzer), die zur Auswertung von Personalisierungsregeln ebenfalls über die Schnittstelle `ProfileIO` ausgelesen werden können, andererseits bestehende reale Profile um Attribute mit statistischen Informationen erweitert werden können (z. B. Clusterzugehörigkeit). In beiden Fällen ist die Entstehung von Verklemmungen beim parallelen Zugriff auf die Profildaten durch Profiling Server und Analyzer zu verhindern.

Darüber hinaus sollte der Analyzer über eine Ablaufsteuerung verfügen, die es erlaubt, Analyseläufe automatisiert in Zeiten geringer Nutzung der Website durchzuführen.

#### 4.2.3.4 Import-Export Manager

Zur Ermöglichung des Imports von Daten in die Profildatenbank und Exports zu anderen Systemen sind entsprechende Ablaufmechanismen vorzusehen, zu deren Konfiguration und Steuerung der Import-Export Manager dient. Auf diese Weise können automatische Datenspiegelungen nach oder von CRM-, Call Center- oder Vertriebsanwendungen durchgeführt werden, um die in Kapitel 2.1.4 geforderte Integration mit Profildaten anderer Kommunikationskanäle zu erreichen.

Zu diesem Zweck greift der Import-Export Manager ebenfalls auf die Objektstruktur des Profilschemas zu, anhand derer leicht eine Visualisierung dessen auf der Client-Oberfläche erstellt werden kann, die eine einfache Selektion der zu importierenden oder exportierenden Datenfelder erlaubt. Die Bereitstellung eines zusätzlichen Regelmechanismus zur verfeinerten Auswahl und Manipulation der zu transferierenden Daten ermächtigt dieses Werkzeug in hohem Maße.

Beispielhaft sei der automatische Export der Stammdaten aller Kunden einer Website zu einer CRM-Anwendung genannt, deren Umsatz einen definierten Schwellwert überschreitet, um diese anschließend über alle Kommunikationskanäle des Unternehmens als bevorzugt zu behandelnde A-Kunden anzusprechen.

Zyklische Datenimporte können dagegen aus einer Anwendung zur Verwaltung eingehender Kundene-Mails stattfinden, um zumindest Anzahl und Zeitpunkte dieser Kontakte im Bereich *Interaction* eines Profils einzutragen.

Als universelles Austauschformat bietet sich XML an. Damit lassen sich, wie in Abschnitt 4.2.3.1 bereits erwähnt, nicht nur Profilschemata und Klassifikationen, sondern auch die eigentlichen Profile in Dateien speichern, sofern ein geeigneter XML-Namespace definiert wurde. Da XML auf der Verwendung eines einfachen Zeichensatzes basiert, eignet sich dieses Format insbesondere für einen unkomplizierten Austausch dieser Informationen mit anderen Anwendungen per Netzwerk oder auf Dateisystemebene.

### 4.2.3.5 Rule Editor

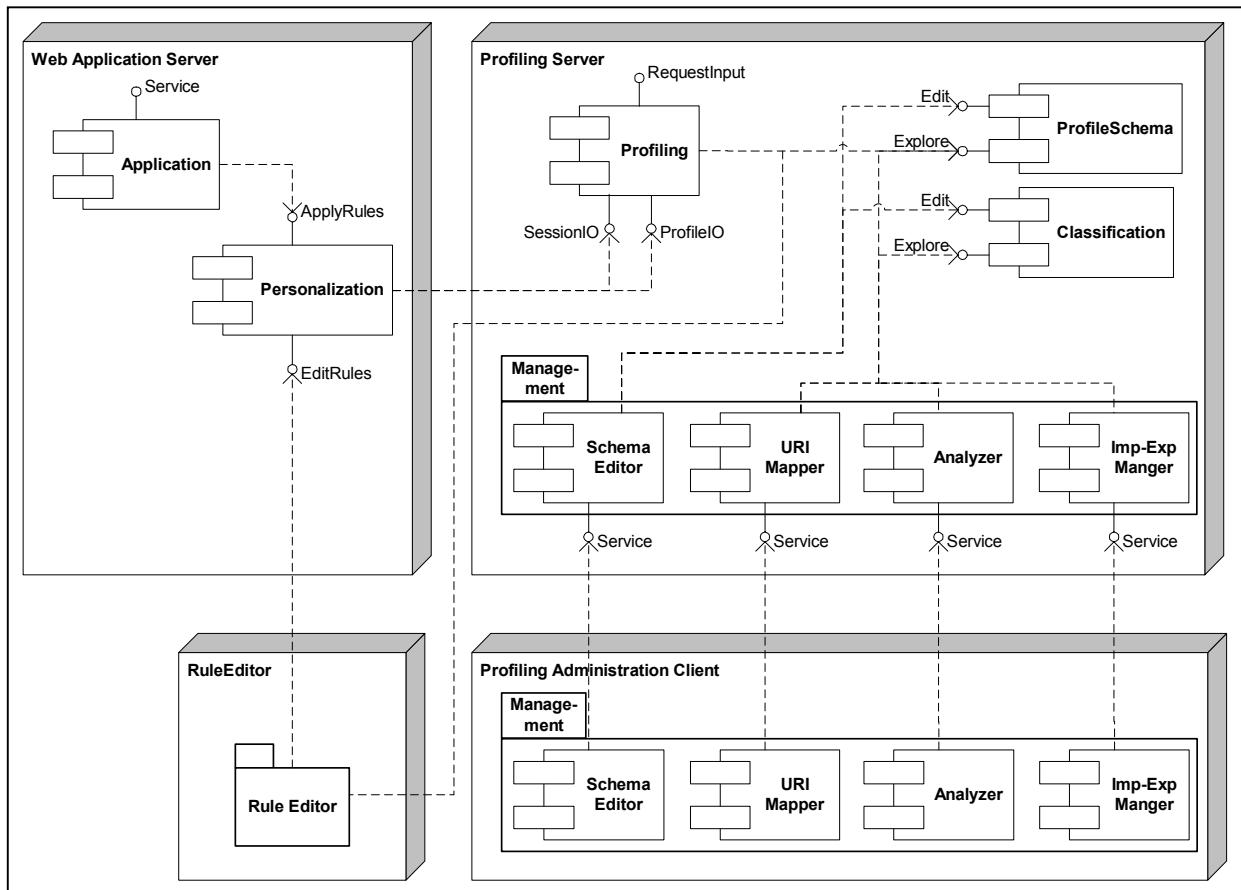
Abschließend wird kurz erörtert, dass ein eigener Rule Editor zum Erstellen und Modifizieren von Personalisierungsregeln als Bindeglied zwischen der Webapplikation und dem zugrunde liegenden Profilingssystem eine sinnvolle Ergänzung darstellt, um bei großen, sich häufig ändernden Online-Angeboten ein hohes Maß an Flexibilität zu erreichen.

Damit gehört ein solches Werkzeug nur noch sehr bedingt zu einem Referenzmodell für Profiling, denn es besteht eine große Abhängigkeit zur basierenden Webanwendung, deren Notwendigkeit zur Regelauswertung zwecks Personalisierung zur Folge hat, dass eine sehr enge Kopplung zwischen Webapplikation und Regelmechanismus notwendig ist. Auf der anderen Seite ist zur Erstellung von Personalisierungsregeln das Wissen um die hierfür zur Verfügung stehenden Daten erforderlich, wozu im vorliegenden Referenzmodell der einfache Zugriff auf die Objektstruktur des Profilschemas und einzelner Profile über definierte Schnittstellen möglich ist.

Insgesamt ist ein bedeutsamer Vorteil zu erwarten, wenn mithilfe eines Rule Editor eine Separation von Profiling, Geschäftslogik und Personalisierungsregeln durchgeführt wird, insbesondere ist ein 'hartes Codieren' dieser Regeln in die Webapplikation zu vermeiden.

#### 4.2.4 Zusammenfassung und Überblick

Die Verteilung und das Zusammenspiel der verschiedenen Komponenten des bisher entworfenen Modells werden in Abbildung 26 veranschaulicht.



**Abbildung 26: Komponenten-/Verteilungsdiagramm Gesamtsystem**

Die in Abbildung 13 vorgestellten elementaren Objekte für Profiling werden zu einer gleichnamigen Komponente im Profiling Server zusammengefasst, welche nach außen die bereits erörterten Schnittstellen `RequestInput`, `SessionIO` und `ProfileIO` anbietet.

Sowohl `RequestInput` als auch die hier `Service` genannte Schnittstelle des zugrunde liegenden Web Application Servers werden dabei vom hier nicht dargestellten Request Interceptor angesteuert.

Die beiden Ableitungen des Kompositum-Musters, `ProfileSchema` und `Classification`, werden als eigene Komponenten dargestellt, da sowohl deren Schnittstellen zum Editieren (`Edit`) und insbesondere die Schnittstellen zum Auslesen der Struktur (`Explore`) von vielen anderen Systembestandteilen genutzt werden.

Dazu gehören vor allem die Serverkomponenten der Administrationswerkzeuge des Systems, denen korrespondierende Client-Komponenten eins zu eins gegenüberstehen, welche die Nutzerschnittstelle für die Anwendung des Profiling-Systems bilden.

Server- und Client-Komponenten der Administrationswerkzeuge sind jeweils zu Paketen zusammengefasst, da sie aufgrund ihrer systemtechnischen Verwandtschaft weitere,

untergeordnete Komponenten gemeinsam nutzen können, vor allem zur Netzwerkkommunikation zwischen Client und Server sowie für die grafische Nutzeroberfläche.

Die zur Personalisierung nutzbaren Schnittstellen `SessionIO` und `ProfileIO` werden bei der in Abbildung 26 vorgestellten Systemarchitektur nicht von der Webapplikation direkt verwendet, sondern mittels einer Personalisierungskomponente, deren Regelwerk durch den in Kapitel 4.2.3.5 vorgeschlagenen Rule Editor vorgegeben wird.

Aus diesem Entwurf eines Modells für Profiling ergibt sich folgender grober Ablauf zur Einführung eines darauf basierenden Gesamtsystems:

1. Es müssen eine oder mehrere für die Anwendungsdomäne geeignete semantische Klassifikationen ausgewählt und importiert bzw. entwickelt werden.
2. Unter Verwendung dieser Klassifikationen wird ein Profilschema als Schablone der anzulegenden Nutzerprofile erstellt.
3. Es erfolgt eine initiale, in möglichst hohem Maße automatisierte Zuweisung semantischer Klassifikationsschlüssel zu den abrufbaren URLs und eine Modellierung der durchführbaren Transaktionen sowie die Definition aller daraus abzuleitenden Profilerweiterungen.

## 5 Instanziierung des Modells auf Anwendungsbeispiele

Aufbauend auf dem in Kapitel 4 erstellten Referenzmodell für Profiling bei e-Business-Websites wird nun zu dessen Evaluation der mögliche Einsatz davon abgeleiteter Systeme bei zwei existierenden Internetanwendungen beschrieben. Dies geschieht ausführlich am Beispiel einer Community-Plattform und wird für einen Online-Versicherungsmakler grob skizziert. In beiden Fällen wird von Implementierungsdetails abstrahiert.

### 5.1 Die Community-Plattform c-lounge

Nach einer Vorstellung des zugrunde liegenden Geschäftsmodells erfolgt die Motivation des Einsatzes von Profilingmaßnahmen bei c-lounge, um nach einer Einführung in die bestehende Webapplikation die Adaption und Integration des erstellten Referenzmodells zu beschreiben.

#### 5.1.1 Geschäftsmodell von c-lounge

Die Website [www.c-lounge.com](http://www.c-lounge.com) wird von der c-lounge AG, Berlin, betrieben und bietet eine Plattform für das Erstellen und Betreiben von Internet-Communities.

Damit stellt c-lounge keine einzelne Community dar, die sich einem bestimmten Thema widmet, sondern bildet das Service-Portal für eine Vielzahl unterschiedlicher Communities, die von verschiedenen Nutzern betrieben werden können, indem typische technische Dienste hierfür angeboten werden, die von den einzelnen Betreibern angepasst und verwendet werden können.

Dieses Geschäftsmodell enthält zwei verschiedene Grundlinien:

- a) In der Basisversion steht c-lounge kostenlos jedermann zur Verfügung. Privatpersonen, Interessensgruppen oder Vereine können c-lounge nutzen, um virtuelle Gemeinschaften zu gründen oder für bestehende Gemeinschaften Online-Services wie Nachrichten- oder Diskussionsforen, Votings und Linklisten zu erschließen.

Communities dieser Art gliedern sich in das Portal [www.c-lounge.com](http://www.c-lounge.com) ein und basieren auf einem gemeinsamen Paket grundlegender Funktionalitäten sowie einem einheitlichen Design.

- b) Darüber hinaus kann der Betrieb einer Community, die einen kommerziellen Zweck erfüllt und ggf. andere Ansprüche an Funktionalität und Design hat, gegen Entgelt beauftragt werden. Diese Dienstleistung richtet sich insbesondere an Unternehmen, die beispielsweise ihre eigene Website um eine Community bereichern möchten, jedoch nicht die ressourcenintensive Eigenentwicklung bzw. Softwareauswahl und -konfiguration sowie den Betrieb eines solchen Systems auf sich nehmen möchten.

In beiden Fällen profitieren die Community-Betreiber von einer betriebsfertigen Lösung, die lediglich konfiguriert und freigeschaltet werden muss. Diese Eigenschaft wird von der c-lounge AG als 'one step community building' kommuniziert.

Die zur Deckung der Entwicklungs- und Betriebskosten notwendigen Erlöse sollen im Fall a) durch die Schaltung von Online-Werbung und im Fall b) durch die anfallenden Leistungsentgelte erwirtschaftet werden.

### 5.1.2 Profiling bei c-lounge

Besucher von c-lounge können gemäß der aktuellen Anwendungspraxis zwei Gruppen angehören: den Betreibern von Communities, die für die Einstellung und Pflege von Content sowie weitere administrative Aufgaben zuständig sind, und den Mitgliedern von Communities, die als registrierte Nutzer Zugang zu einzelnen Communities haben.

Nutzer können sowohl mehrere Communities betreiben als auch in verschiedenen Communities Mitglied sein.

Neben der Speicherung dieser Nutzungsverhältnisse, die für den Betrieb unmittelbar notwendig ist, findet bisher weder eine weiterführende Datenerhebung noch –auswertung statt.

Im Hinblick auf die skizzierten Erlösmodelle bei c-lounge ist die Erhebung und Verwendung individueller Nutzerdaten jedoch von großem Interesse.

Im Fall a) besteht die Möglichkeit, durch die Schaltung personalisierter Werbung, die den jeweiligen Präferenzen der Nutzer entgegenkommt, höhere Einnahmen zu erzielen, da die zielgruppengenaue Ansprache potenzieller Konsumenten zu geringeren Streuverlusten führt und derartige Werbepplätze nicht nur im Markt für Online-Werbung durch Werbetreibende höher dotiert werden.

Während in klassischen Medien die Zielgruppengenauigkeit oftmals durch die thematische Spezialisierung des Mediums erreicht wird, z. B. Print-Werbung für Autozubehör in Autofachzeitschriften, bietet das Internet die Möglichkeit, auf 'general-purpose'-Websites fokussierte Werbung zu betreiben, indem im individuellen Kommunikationskontext auf vorhandene Informationen über einzelne Nutzer, deren Profile, zurückgegriffen wird.

Im Fall b) spielt die von einem Unternehmen betriebene Community eine besondere Rolle für die Marktbeobachtung. Neben Vorteilen der zielgenauen Kommunikation und bei der Akquisition bieten Communities Unternehmen "Bessere Möglichkeiten, bestehende Produkte und Dienstleistungen auf die Kunden zuzuschneiden und dadurch im Wert zu steigern: Anbieter die mehr über die bisherigen Geschäftsaktivitäten von Kunden und potenziellen Käufern wissen, weil sie mit ihnen interagieren können, werden die individuellen Kundenbedürfnisse besser verstehen. Wenn die Anbieter diese Informationen konsequent nutzen, Produkte entsprechend zuschneiden und Produkt- und Servicepakete schaffen, können sie sowohl ihren potentiellen Kundenstamm ausbauen als auch höhere Gewinne pro Kunde erzielen" [HaAr97].

Dies setzt jedoch eine umfassende Datenbasis voraus, die es erlaubt, aussagestarke Analysen durchzuführen und Nutzer individuell zu charakterisieren.

In beiden Fällen ist somit der Einsatz eines Profilingssystems deutlich indiziert, wobei es die Flexibilität des in Kapitel 4 erstellten Modells erlaubt, bei der Realisierung bzw. Konfiguration eines darauf basierenden Systems funktionale Schwerpunkte hinsichtlich der beiden Ziele zu setzen.



Da c-lounge zurzeit ausschließlich in der Ausprägung von Geschäftsmodell a) betrieben wird, konzentrieren sich die nachfolgenden Überlegungen für den Einsatz von Profiling hierauf. Der Einsatz für Geschäftsmodell b) setzt unbedingt Wissen über die Motivation des jeweiligen Betreibers und die konkrete Ausgestaltung der Community voraus, ein grober Ausblick auf die potenziellen Möglichkeiten findet sich in Abschnitt 5.1.5.

### 5.1.3 Beschreibung der Webanwendung von c-lounge

Die technische Beschreibung der c-lounge geschieht nachfolgend durch die Darstellung der den Nutzern gebotenen Services und des damit verbundenen Workflows bei der Benutzung der Website sowie durch die Vorstellung der softwaretechnischen Architektur.

#### 5.1.3.1 Workflow für Nutzer

Alle Besucher der Website erhalten auf der einleitenden Portalseite von [www.c-lounge.com](http://www.c-lounge.com) (vgl. Abbildung 27) die Möglichkeit, die vorhandenen Kategorien mit darin enthaltenen Communities zu explorieren, sich als Nutzer registrieren zu lassen oder die von allen Seiten erreichbaren übergeordneten Querschnittsfunktionen aufzurufen (Kontakt, Hilfe, Impressum, AGB).

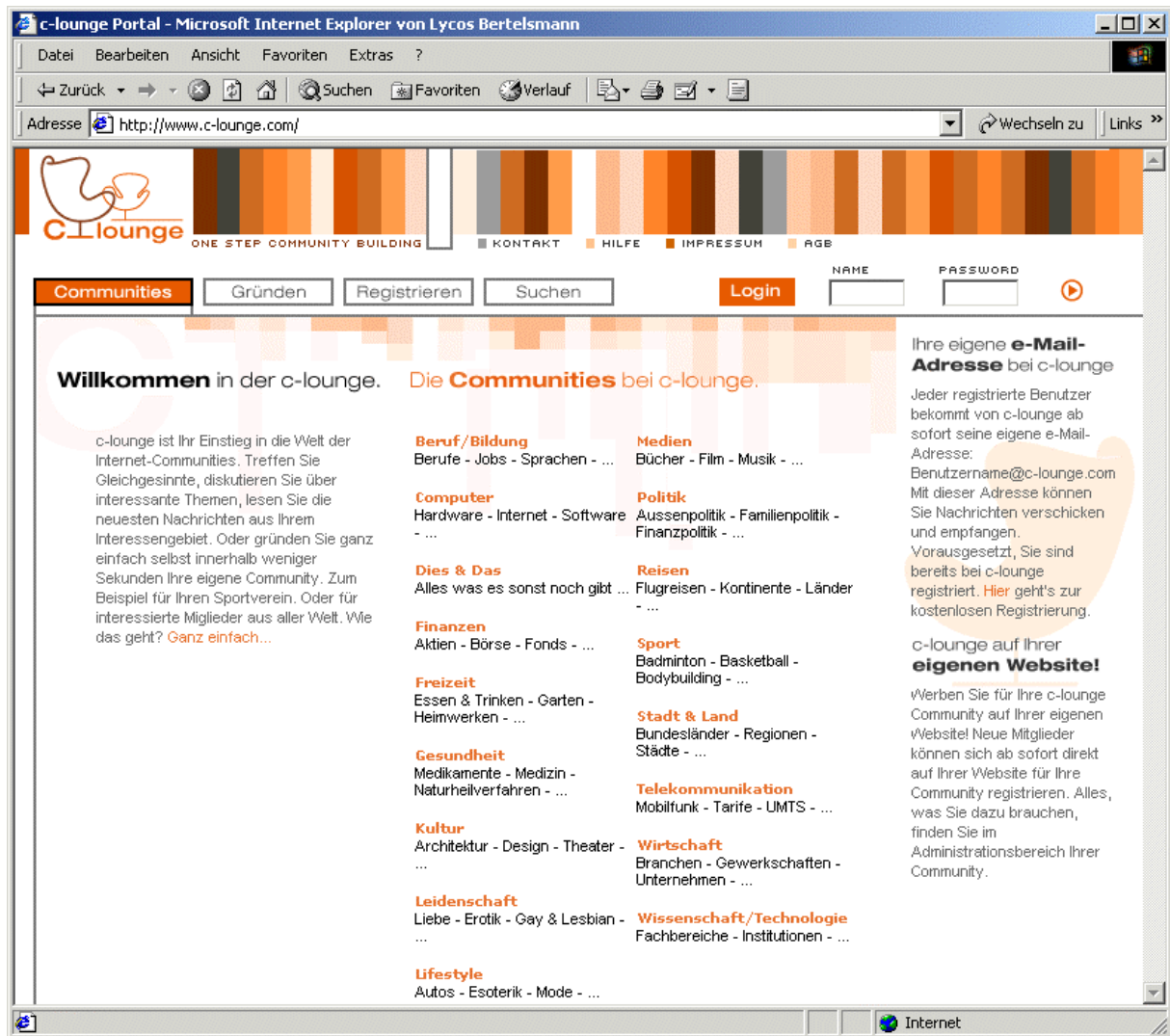


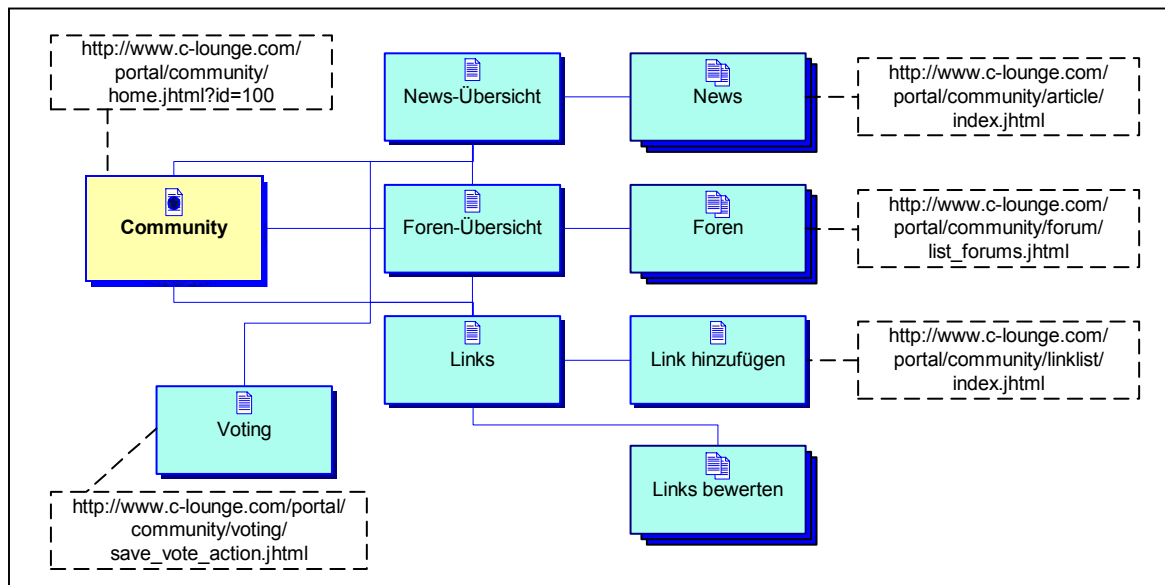
Abbildung 27: Screenshot c-lounge Portal

Registrierte Nutzer erhalten nach erfolgreichem Login eine personalisierte Übersicht über die Communities, bei denen sie Mitglied oder Betreiber sind.

Mitgliedern stehen nach Betreten einer Community vier Bereiche zur Verfügung:

- **Community:** Diese Leitseite erscheint nach Betreten einer Community und enthält neben einer Grafik und einer Begrüßungsmeldung die aktuellsten News-Meldungen sowie ein Voting, sofern der Betreiber diese angelegt hat.
- **News:** Hier werden alle News-Meldungen dieser Community angezeigt. Einzelne Meldungen können von den Mitgliedern bewertet werden, falls der Betreiber diese Funktion beim Einstellen der Nachricht aktiviert hat.
- **Foren:** Eine Tabelle enthält die Übersicht über alle Foren. Wird ein Forum geöffnet, so erscheint der Thread seiner Diskussionsbeiträge, zu denen weitere Äußerungen abgegeben werden können.
- **Links:** Auf dieser Seite erscheinen Links einschließlich Beschreibungen, die verschiedenen Kategorien zugeordnet werden können. Mitglieder und Betreiber können neue Links hinzufügen, danach suchen und diese bewerten.

Abbildung 28 stellt die Sitestruktur im Mitgliederbereich einer Community dar. Die oben erläuterten Seiten sind als Rechtecke dargestellt, die sie verbindenden Links als Linien. Zur Veranschaulichung der von `SemanticsTable` potenziell zu verarbeitenden Informationen sind die jeweiligen URLs in gestrichelten Rahmen neben den einzelnen Seiten vermerkt.



**Abbildung 28: Sitestruktur c-lounge Community-Mitglied**

Betreiber von Communities können darüber hinaus zahlreiche administrative Tätigkeiten vornehmen, die sich in folgende Bereiche gliedern und in Abbildung 29 dargestellt sind:

- **Community:** Hier erfolgt die Einstellung globaler Parameter, z. B. Begrüßungstext, Begrüßungsgrafik, Auswahl des Votings auf der Startseite der Mitglieder, Konfiguration, ob die Community offen oder geschlossen ist, d. h. dass der Betreiber über die Mitgliedschaft von Interessenten entscheiden kann.
- **News:** Erlaubt das Einstellen, Platzieren und Löschen von News-Meldungen.
- **Newsletter:** Hier wird dem Betreiber einer Community die Möglichkeit geboten, eine Mail an alle Mitglieder zu versenden.
- **Foren:** Es können Diskussionsforen zu frei wählbaren Themen angelegt werden. Als Forum wird dabei ein Thread verstanden, innerhalb dessen die einzelnen Postings in einer Baumstruktur eingefügt werden können.
- **Links:** Auf dieser Seite können die Kategorien angelegt und gelöscht werden, innerhalb derer die Mitglieder Links einstellen können.
- **Voting:** Anlegen, aktivieren, deaktivieren und Löschen von Votings. Beim Anlegen eines Votings werden die möglichen Antworten eingegeben.
- **User:** Hier besteht die Möglichkeit, bekannte Interessenten per e-Mail einzuladen, nach eingetragenen Mitgliedern zu suchen, sich eine Liste aller Mitglieder anzeigen zu lassen sowie einzelne Mitglieder aus der Community auszuschließen, was u. a. bei Missachtung der c-lounge-Benutzungsregeln geboten ist.

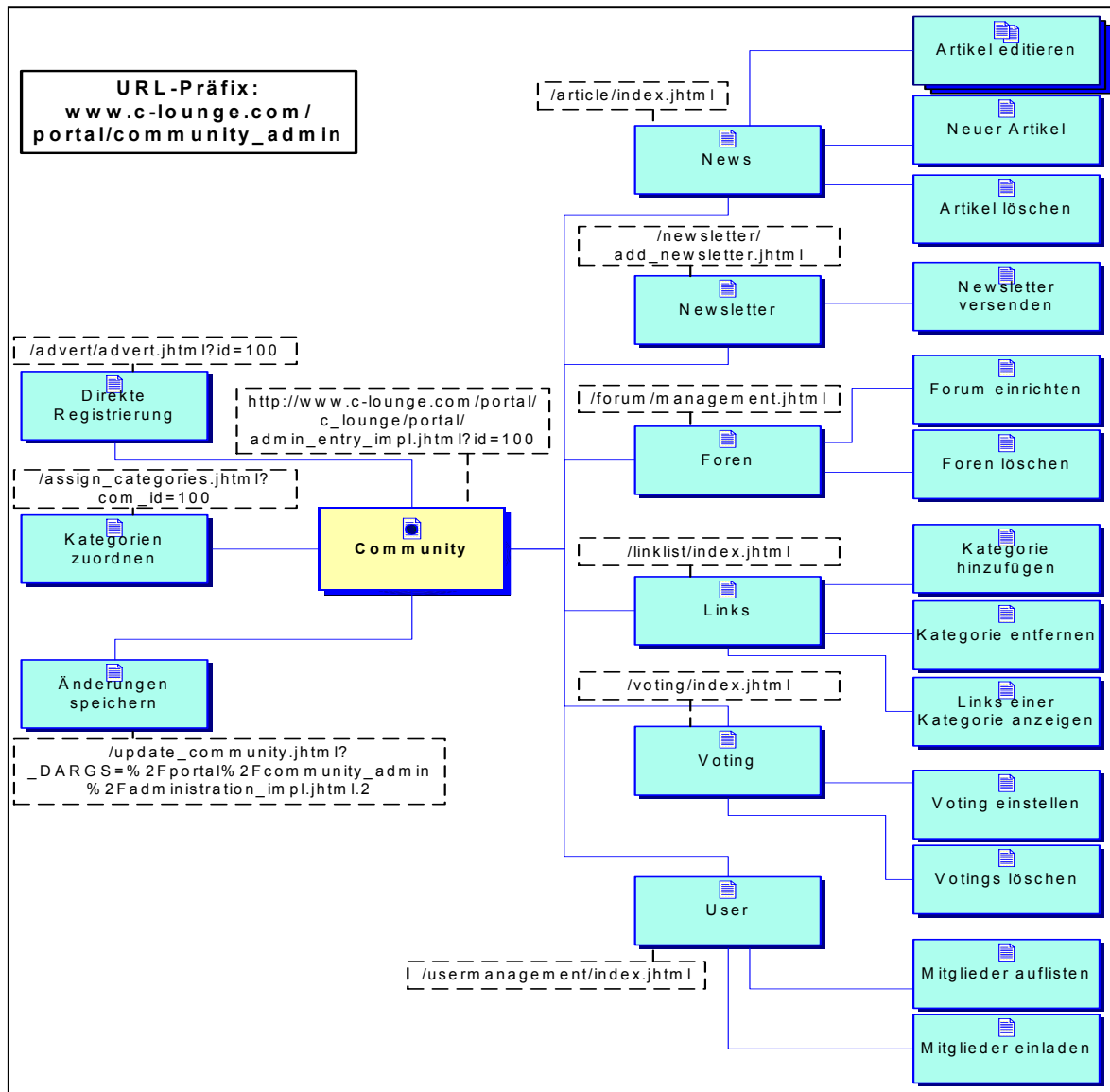


Abbildung 29: Sitestruktur c-lounge Community-Betreiber

### 5.1.3.2 System- und Softwarearchitektur

Der Betrieb der Website www.c-lounge.com basiert auf dem Einsatz des Applikationsservers Dynamo von ATG und des Webserver Apache auf einem Sun-Rechner mit dem Betriebssystem Solaris in Verbindung mit einer Datenbank des Typs Oracle 8i auf einem dedizierten Datenbankserver, der unter Linux betrieben wird.

Die Anwendung ist durch eine Reihe in verschiedene Pakete unterteilte Java-Klassen und Servlets realisiert, deren Funktionalität aus sogenannten JHTML-Seiten aufgerufen wird. Der Abruf einer solchen Seite per HTTP-Request führt ähnlich der Arbeitsweise von JSP zur Instanziierung und Aktivierung der darin eingebetteten Servlets durch den Applikationsserver. Die Servlets führen unter Zugriff auf die zugrunde liegende Datenbank spezifische Verarbeitungsschritte aus und generieren Ausgaben, die in das Grundgerüst der Seite integriert und als HTML an den Client zurückgeschickt werden.

Der Applikationsserver erleichtert dabei die Anwendungsentwicklung und erhöht die Performanz durch bereitgestellte Mechanismen u. a. für das Session-Management sowie das Caching und die Wiederverwendung von Servlets in der Laufzeitumgebung.

Gemäß der Rekonstruktion einer Beschreibung der System- und Softwarearchitektur in [Hand01] gliedern sich die fachlichen Komponenten der c-lounge-Anwendung in drei Hauptpakete:

- application: Diese Ansammlung von Java-Servlets und -Klassen bildet in Kombination mit den JHTML-Seiten den c-lounge-spezifischen Teil der Applikation, was bereits durch die Ähnlichkeit der in ihr befindlichen Komponenten (vgl. Abbildung 30) und den für Nutzer möglichen Aktionen aus Abbildung 28 ersichtlich wird.
- core: Hier befinden sich allgemeinere Service-Komponenten, welche grundlegende Funktionalitäten der Anwendung als Dienst für 'application' erbringen. Dazu gehören Aufgaben des Nutzer-, Gruppen- und Rechte-Managements, aber auch für verschiedene andere Anwendungen einsetzbare Komponenten zur Verwaltung von redaktionellen Artikeln, Bulletin-Boards oder Abstimmungen.
- system: Diese Basisdienste ermöglichen allen anderen Komponenten übergreifend Funktionen für das Monitoring und Logging von Aktivitäten und insbesondere eine einheitliche Schnittstelle zur Datenbank.

Diese grobe Dreiteilung ist in Abbildung 30 als Komponentendiagramm dargestellt.

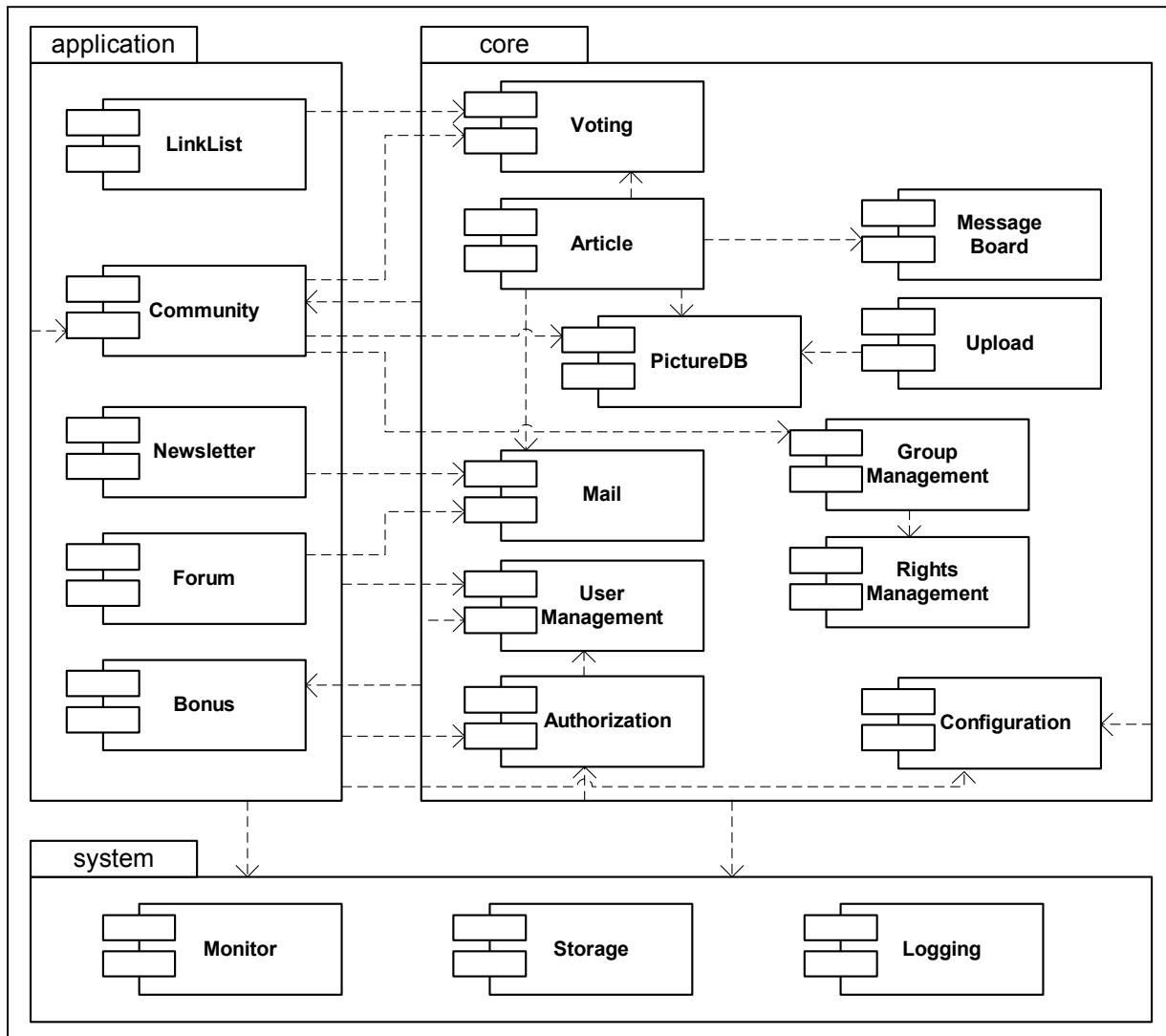


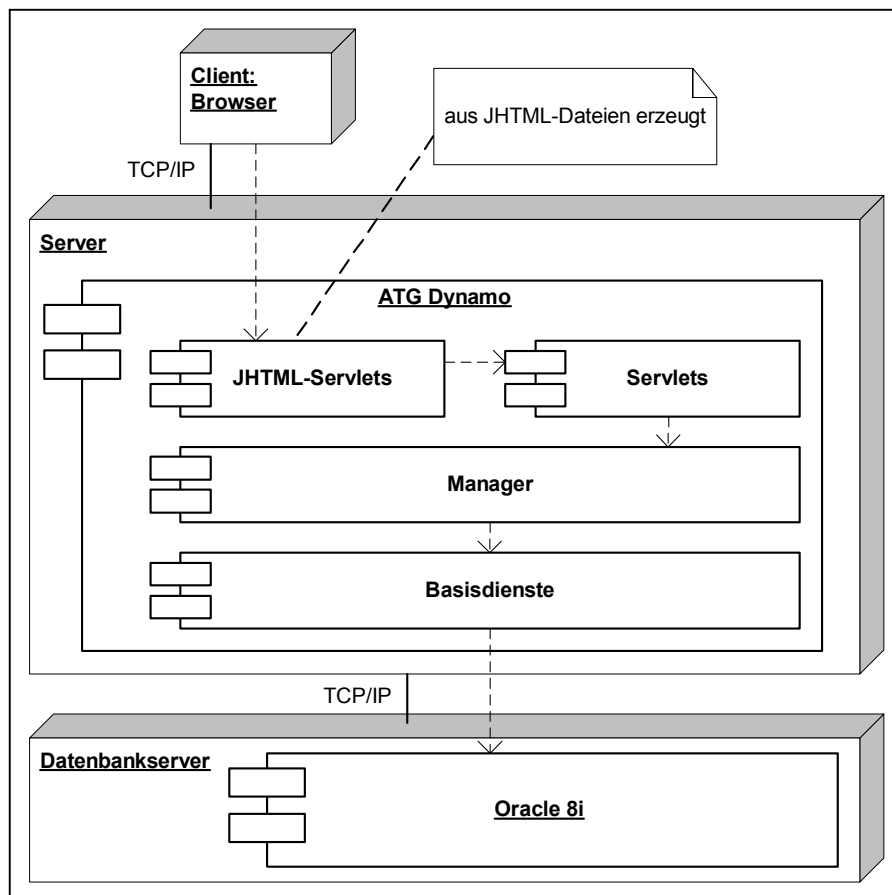
Abbildung 30: Komponentendiagramm c-lounge [Hand01]

Die enge Verflechtung zwischen den Komponenten wird innerhalb des 'core'-Pakets am Beispiel von 'Article' deutlich: da das Einstellen einer Textnachricht mit einer Bewertung und einem Diskussionsforum hierzu verknüpft werden kann, ist eine Verwendung von 'Voting' und 'Message Board' vorgesehen. Darüber hinaus können Nachrichten durch Bilder ergänzt werden ('PictureDB') und der Versand einer e-Mail an den Autor ist möglich ('Mail').

Aus softwaretechnischer Sicht werden die meisten dieser fachlichen Komponenten durch verschiedene Einzelkomponenten realisiert:

- Servlets zur Verarbeitung der Client-Requests, die aus JHTML-Seiten aufgerufen werden, sofern sie einer grafischen Oberfläche in Form von HTML bedürfen,
- Manager-Klassen, welche die eigentliche Verarbeitungslogik enthalten und insbesondere für die Manipulation von Daten zuständig sind, sowie
- Basisdienst-Klassen, die beispielsweise den Datenbankzugriff realisieren.

Mit dieser Dreiteilung der Mittelschicht des Gesamtsystems können Zuständigkeiten verteilt und auf bestimmte Stellen konzentriert werden, so ist z. B. kein direkter Zugriff auf die Datenbank durch Servlets notwendig, Manager-Klassen brauchen nicht die Präsentation der von ihnen verarbeiteten Daten berücksichtigen und technische Details des Datenbankzugriffs sind in den Basisdiensten gekapselt. Diese Architektur ist in Abbildung 31 schematisch beschrieben.



**Abbildung 31: Komponenten-/Verteilungsdiagramm c-lounge**

#### 5.1.4 Einsatz des Referenzmodells für Profiling bei c-lounge

Nachfolgend wird eine Möglichkeit für Profiling bei c-lounge aufgezeigt, indem zunächst die fachlichen Grundlagen hierfür beschrieben werden und anschließend die system- und softwaretechnische Integration des in Kapitel 4 erstellten Referenzmodells entworfen wird.

##### 5.1.4.1 Fachliche Grundlagen

Ausgehend von der in Abschnitt 5.1.2 skizzierten Motivation für Profiling bei c-lounge ist es ausreichend, zwischen Betreibern und Mitgliedern von Communities zu unterscheiden. Charakterisierend für beide Nutzergruppen sind die Communities, die jeweils betrieben oder besucht werden. Deshalb eignet sich als zu verwendende Klassifikation für die online abrufbaren Inhalte die vorhandene Kategorisierung der virtuellen Gemeinschaften, wie sie in Abbildung 32 dargestellt wird. Die erste Ebene dieser Kategorisierung ist auch in Abbildung 27 zu erkennen.

Um neben der bloßen Zugehörigkeit eines Nutzers zu einer Community, sei es als Betreiber oder Mitglied, auch das Maß seiner Aktivität widerspiegeln zu können, wird vorgeschlagen, Pageviews innerhalb des Nutzerprofils zu erfassen und dabei den einzelnen Communities zuzuordnen.

Abbildung 33 zeigt ein Profilschema für den Einsatz bei c-lounge, das den o. g. Überlegungen Rechnung trägt. Zur Implementierung wird das Analysemuster 'Kompositum' verwendet, wie in Kapitel 4.2.1.3 dargelegt. Einzelne Instanzen von `Leaf` sowie die jeweils notwendigen in der Baumstruktur oberhalb befindlichen `Composition`-Objekte werden erst angelegt, wenn ein Nutzer Mitglied oder Betreiber einer Community wird. Jedes `Leaf`-Objekt enthält einen ganzzahligen Wert, der die Anzahl der Pageviews innerhalb der repräsentierten Community enthält, `Composition`-Objekte enthalten den kumulierten Wert aller Kindknoten.

Die durch dieses Vorgehen erreichten Profile sind gut geeignet für ihren Verwendungszweck der personalisierten Online-Werbung, da die Klassifikation zur Speicherung der Nutzerpräferenzen den in c-lounge enthaltenen Content per Definition vollständig abdeckt und zudem in eine Hierarchie jeweils disjunkter Mengen unterteilt. Dadurch werden mögliche Nutzerinteressen ausreichend genau beschrieben und die Klassifikation unterliegt keinen Veränderungen, abgesehen von gelegentlichen, schrittweisen Erweiterungen.

Beim Zugriff auf das Profil können die Bereiche in einfacher Weise exploriert werden, innerhalb derer ein Nutzer Mitglied oder Betreiber einer Community ist. Die in jeder Kategorie gespeicherte Anzahl seiner Pageviews bietet ein verlässliches Maß zur Differenzierung von häufig und selten besuchten Communities und Bereichen und erlaubt darüber hinaus einen Vergleich der Aktivität zwischen verschiedenen Nutzern.



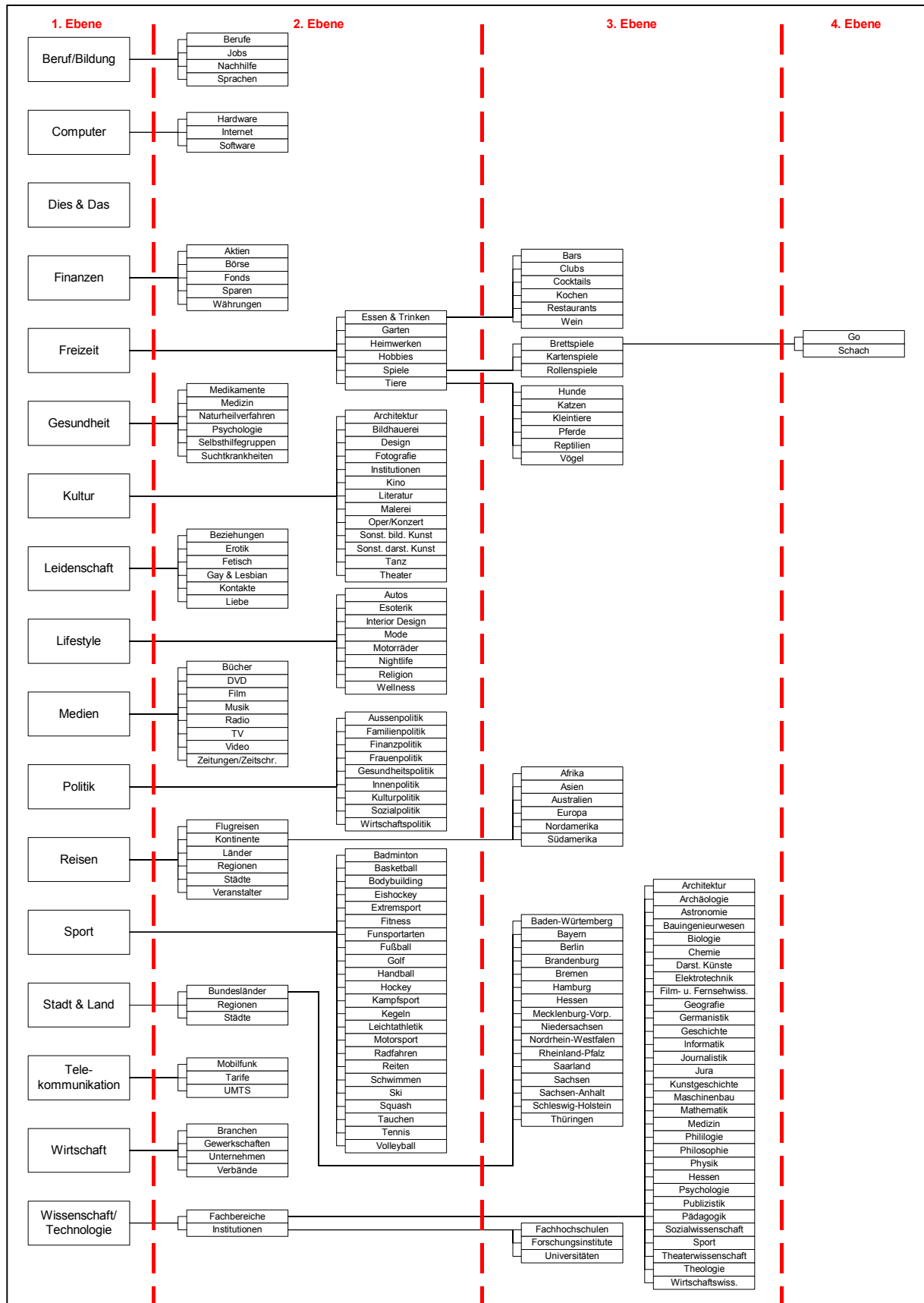


Abbildung 32: Kategorisierung der c-lounge-Communities

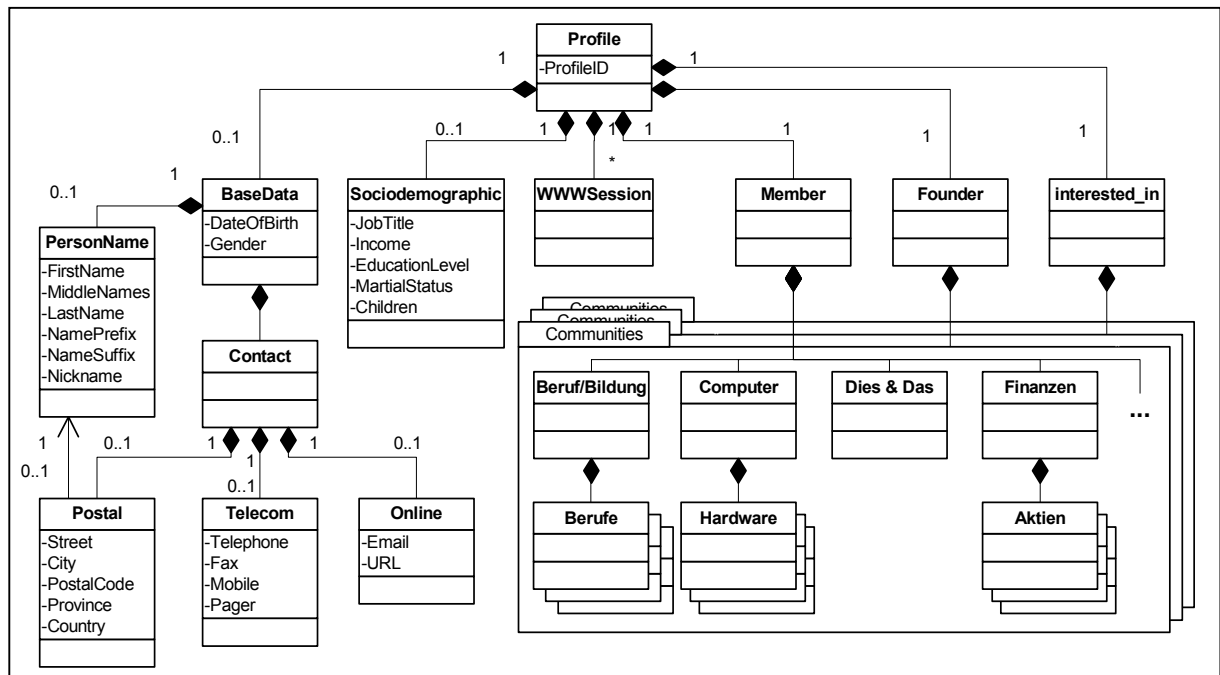


Abbildung 33: Profilschema c-lounge

#### 5.1.4.2 System- und softwaretechnische Integration

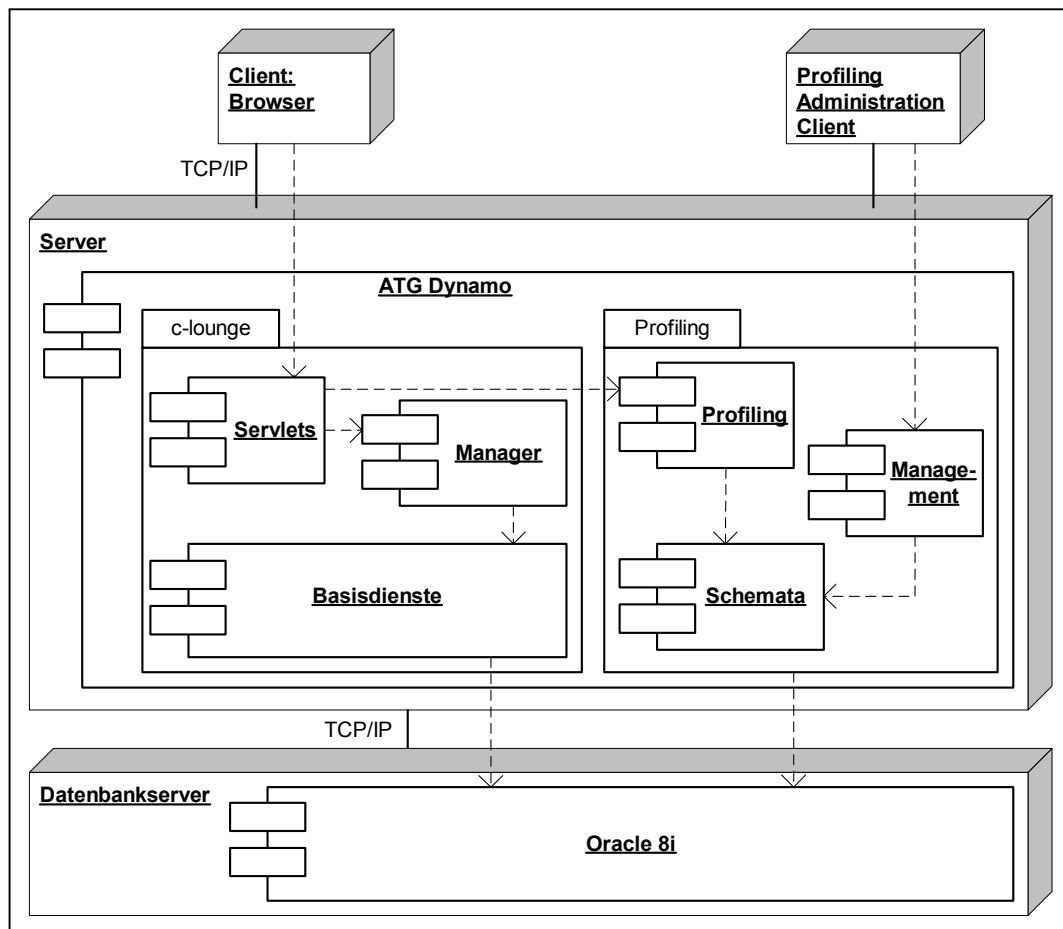
Zur Integration des in Kapitel 4 erstellten Objektmodells in dieses Anwendungsbeispiel wird die Implementierung der oben entworfenen Klassen in Java empfohlen, welche zu einem Subsystem zusammengefasst und neben den Hauptkomponenten von c-lounge mithilfe des ATG Dynamo betrieben werden.

Diese Architektur ist in Abbildung 34 zusammenfassend dargestellt, wobei die im Paket 'Profiling' dargestellten Komponenten jene aus Abbildung 26 enthalten, deren einzelne Klassen in Kapitel 4.2 detailliert beschrieben wurden: die Komponente 'Profiling' enthält insbesondere die Klassen `SessionManager` und `ProfileManager`, welche die Schnittstellen `RequestInput`, `SessionIO` und `ProfileIO` realisieren; 'Schemata' enthält die hierarchischen Objektinstanzen `ProfileSchema` sowie `Classification`, und 'Management' dient als serverseitige Schnittstelle für den Profiling Administration Client.

Implementierung und Test dieser Klassen kann vollständig unabhängig vom Betrieb von c-lounge erfolgen und auch der Einsatz eines auf diese Weise entwickelten Profilingssystems erfolgt weitestgehend parallel zur bestehenden Anwendung, jedoch innerhalb derselben Applikationsserver-Umgebung.

Den zunächst einzigen Kontaktpunkt zwischen beiden Anwendungen bildet der nach den Überlegungen aus Abschnitt 4.2.1.1 erforderliche Request Interceptor, welcher durch die in Dynamo integrierte Funktionalität besonders leicht realisiert werden kann, da alle HTTP-Requests von Clients im Applikationsserver an zentraler Stelle von einer sogenannten 'Servlet Pipeline' verarbeitet werden (vgl. [ATG00]). Diese Verarbeitungskette lässt sich durch den Einschub eines eigenentwickelten Servlets verlängern, welches die mitgelieferte Klasse `atg.servlet.pipeline.InsertableServletImpl` erweitert, die bereits alle durch das Java-Servlet-Interface geforderten Methoden implementiert. Durch Überschreiben

der `service()`-Methode kann beliebige eigene Funktionalität in die Verarbeitung von HTTP-Requests integriert werden, hier also das Ansprechen der Profilingkomponente mit den aus dem Request ausgelesenen Daten. Dabei ist lediglich zu beachten, dass das selbst entwickelte Servlet am Ende der Servlet Pipeline aufgerufen wird, nachdem das Dynamo-'SessionServlet' eine Zuordnung des Requests zu einer Nutzersitzung vorgenommen hat und alle damit assoziierten Daten an das Java-Objekt `request` gebunden wurden.



**Abbildung 34: Komponenten-/Verteilungsdiagramm c-lounge mit Profiling**

Aus `request` können nun leicht die Session-ID sowie die abgerufene URL, alle per `GET` oder `POST` übermittelten Daten sowie Cookies ausgelesen und dem Profilingssystem, ggf. umformatiert, übergeben werden.

Da dieser Datenaustausch asynchron erfolgen sollte, damit Webapplikation und Profilingkomponente parallel arbeiten können, wird zu dessen Umsetzung die Verwendung des Java-Event-Mechanismus vorgeschlagen: statt Aufruf einer Funktion `invokeRequest()` (vgl. Kapitel 4.2.1 ff.) der Schnittstelle `RequestInput` löst das in die Pipeline integrierte Servlet ein selbst definiertes Event aus, auf das der `SessionManager` als `EventListener` reagiert. Dabei können alle zur Profilbildung erforderlichen Daten als Objekt einer von `java.util.EventObject` abgeleiteten Klasse mitübergeben werden.

Die Vergabe der notwendigen Session-ID wird durch Dynamo sichergestellt, indem der Java-typische Session-Tracking-Mechanismus implementiert wird, welcher wie in Kapitel 3.1.2 ff. erwähnt zunächst die Identifizierung der Nutzer per Cookie versucht, um bei Nichterfolg URL-Rewriting zu betreiben [ATG00]. Diese Funktionalität entbindet Softwareentwickler von der eigenständigen Lösung dieser Aufgabe und ermöglicht im vorliegenden Fall die leichte Identifikation des zu einem Nutzer gehörenden Profils. Da Nutzer im Profilingssystem während der Dauer ihrer Session mit der gleichen SID wie in der c-lounge-Applikation referenziert werden, können die Komponenten des 'application'-Paketes beispielsweise zwecks Personalisierung in einfacher Weise auf Attribute von Nutzerprofilen zugreifen, indem sie die Schnittstellen `SessionIO` und `ProfileIO` ansprechen.

Zur Veranschaulichung des Informationsflusses und der Abläufe wird nachfolgend das Beispiel eines Nutzers betrachtet, der c-lounge ansteuert, einige Community-Kategorien durchsucht, sich am System anmeldet und eine von ihm betriebene Community administriert (Abbildung 35).

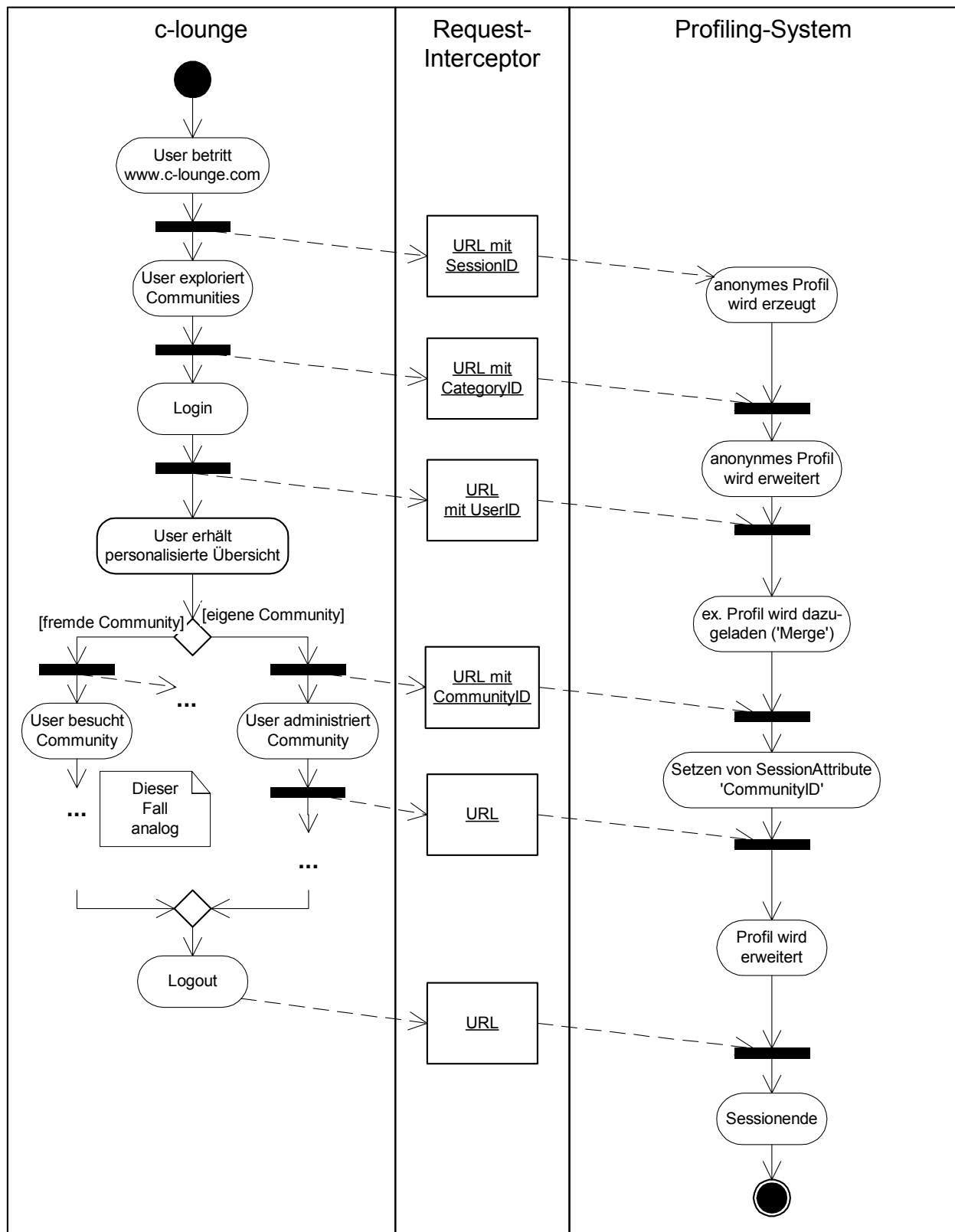


Abbildung 35: Aktivitätsdiagramm c-lounge-Profiling

1. Der Nutzer betritt die Community-Plattform durch Eingabe der URL 'www.c-lounge.com' in seinen Browser. Der Applikationsserver beantwortet den Request und setzt ein Cookie bzw. codiert die Session-ID in das HTML der Antwortseite.

Der Request Interceptor spaltet den Request ab leitet ihn an die Profilingkomponente weiter, nachdem die vom Applikationsserver generierte SID an das `request`-Objekt gebunden wurde.

(Beispiel bei URL-Codierung: '\$sessionId\$HMxKMKIAAaPYFQ53CVISFEY').

Die Profilingkomponente erzeugt ihr eigenes Session-Objekt und ein korrespondierendes anonymes Profil.

2. Der Nutzer durchsucht verschiedene Kategorien der Community-Plattform.  
Die dabei abgespaltenen URLs haben die Form 'www.c-lounge.com/portal/c\_lounge/browse\_community\_category.jhtml?cat\_id=121', wobei in `SemanticsTable` nur nach dem Muster 'portal/c\_lounge/browse\_community\_category.jhtml' gesucht werden muss und die 'cat\_id=' nachgestellte Zahl die eindeutige Nummer der gewählten Kategorie darstellt.  
Dies führt im Profil des Nutzers zu einem Eintrag der Art 'Profile(<PID>).interested\_in.<cat\_id>', wobei gemäß einer 'create-and-count'-Semantik der Klassifikation der die Kategorie repräsentierende Knoten an der entsprechenden Stelle im Profil angelegt bzw. dessen Zähler für Pageviews inkrementiert wird (vgl. Abbildung 33).
3. Der bereits bei c-lounge registrierte Nutzer füllt das Login-Formular mit seinem Nutzernamen und Passwort, um sich am System anzumelden.  
Die übertragene URL enthält beide Angaben und löst in der Profilingkomponente die Anweisung `loadProfile` aus, woraufhin das vorhandene Profil aus der Datenbank geladen werden kann und eine implizite 'Merge'-Operation beide Profile verschmilzt.
4. Unter den gegebenen Interaktionsmöglichkeiten wählt der Nutzer die Administration einer von ihm betriebenen Community aus.  
Die URL entspricht der Form 'http://www.c-lounge.com/portal/c\_lounge/portal/admin\_entry\_impl.jhtml?id=100', wobei der Wert hinter 'id=' die Community bezeichnet und per `writeSessionAttribute` in der Profilingkomponente gespeichert werden muss, da er bei nachfolgenden Administrationsschritten nicht mehr mit der URL übertragen wird.
5. Der Nutzer führt eine Reihe von Administrationsschritten innerhalb seiner Community durch, deren spezielle Bedeutung gemäß der am Anfang von Abschnitt 5.1.4.1 gesetzten Ziele nicht von Interesse ist.  
Zur Registrierung der Aktivität reicht es aus, das Muster 'portal/community\_admin/' in den URLs der Requests zu erkennen und dem entsprechend den Profileintrag zur in Schritt 4 gespeicherten Community zu modifizieren, in diesem Fall also innerhalb des Präferenzbereichs 'Founder' des Profils.
6. Der Nutzer betätigt den Logout-Button, was zur Ausführung der Anweisung `stopSession` in der Profilingkomponente und der damit einhergehenden persistenten Speicherung des Nutzerprofils führt.

Die Schritte 4 und 5 verlaufen völlig analog, falls der Nutzer statt der Administration einer eigenen den Besuch einer fremden Community vollzieht, bei der er Mitglied ist. Alle dabei durchführbaren Aktionen werden durch URLs gekennzeichnet, die das Muster

'portal/community/' enthalten. Die Registrierung der Pageviews geschieht in diesem Fall unterhalb von 'Member' im entsprechenden Nutzerprofil.

Dieses Beispiel belegt damit einerseits die prinzipielle Einsetzbarkeit des in Kapitel 4 erstellten Referenzmodells für Profiling, zeigt andererseits aber auch, dass bei einer systematischen Architektur der zugrunde liegenden Webanwendung der in Abschnitt 4.2.1.1 bemängelte Aufwand zur Übertragung der Applikationslogik trotz komplexer Interaktionsmöglichkeiten in engen Grenzen gehalten werden kann. So ist die Menge der für den Einsatz bei c-lounge in `SemanticsTable` zu hinterlegenden URLs und Aktionen unerwartet gering und kann manuell gepflegt werden.

Die Datenhaltung der Profilingkomponente kann mithilfe der gleichen Datenbank erfolgen, die von der c-lounge-Anwendung benutzt wird. Dazu ist ein objekt-relationales Mapping insbesondere der Profildaten notwendig, wofür eine mögliche Vorgehensweise bereits in Abschnitt 4.2.3.1 skizziert wurde.

Ein zusätzlicher Vorteil kann durch die gemeinsame Nutzung von Daten entstehen, so könnte beispielsweise innerhalb der Profilingkomponente die Auflösung der Community-IDs zu bezeichnenden Namen durch Zugriff auf die entsprechende Datenbanktabelle der c-lounge erfolgen.

### 5.1.5 Bewertung und Ausblick

Abschließend erfolgt ein kurzer Ausblick auf die notwendigen Schritte zur Realisierung des Referenzmodells für Profiling und Aspekte seiner Nutzung im Kontext von c-lounge.

Wie schon in Abschnitt 4.2.4 skizziert, ist nach einer notwendigen Verfeinerung und Implementierung des Modells sowie der bereits erfolgten Auswahl einer geeigneten Klassifikation und dem Entwurf eines Profilschemas der systematischen und vollständigen Zuweisung semantischer Schlüssel bzw. nutzbringender Instruktionen in `SemanticsTable` besondere Bedeutung zuzumessen. Im vorliegenden Fall erscheint es praktikabel, die erforderlichen Einträge anhand der bei c-lounge möglichen Interaktionsabläufe für Nutzer mithilfe eines Werkzeugs manuell vorzunehmen. Obwohl diese Aufgabe durch eine gute Strukturierung der Webapplikation und der damit einhergehenden URLs erleichtert wird, ist zu beachten, dass alle relevanten URLs erfasst werden und keine 'Schlupflöcher' innerhalb der Nutzungsmöglichkeiten der Website verbleiben. Zur Überprüfung dieses Sachverhalts ist wiederum Werkzeugunterstützung wünschenswert; zumindest kann ein Abgleich der im Profilingssystem erfassten URLs mit den Log-Dateien des Applikationsservers dazu beitragen, nicht klassifizierte Links zu entdecken.

Von besonderem Interesse ist schließlich der Nutzen, den der Einsatz der Profilingkomponente für c-lounge erbringt. Anhand der Ausrichtung der hier beschriebenen Maßnahmen an der grundlegenden Zielsetzung der Einblendung personalisierter Werbung ist der hierfür angestrebte Vorteil erreicht: eine c-lounge-Erweiterung zur Ansteuerung eines *Adservers* kann leicht auf das Profil eines Nutzers zugreifen und die Belegung charakterisierender Attribute überprüfen. Diese beziehen sich nicht nur auf soziodemografische Daten, soweit vorhanden, sondern auch auf Präferenzen des Nutzers,

die durch seine Beziehung zu Communities als Interessent, Mitglied oder Betreiber ausgedrückt werden.

Neben recht groben Zuordnungen, wie beispielsweise Werbung für Rasierapparate anhand Geschlecht und Alter, können auch feinspezifische Affinitäten erschlossen werden: Werbung für Wein und Spirituosen für Nutzer der Community-Kategorien 'Wein' und 'Cocktails', Werbung für Fachliteratur für nahezu alle Bereiche (beispielhaft 'Sprachen', 'Hardware', 'Kochen', 'Theater', 'Kunstgeschichte'), usw.

Zum automatisierten Abgleich von Nutzerprofil und Werbebotschaft können auch die verfügbaren Banner einzelnen Elementen der bei c-lounge verwendeten Klassifikation zugeordnet werden. Hierfür bietet sich die Nutzung der Schnittstelle `explore` der Komponente `Classification` durch c-lounge an (vgl. Abbildung 26).

Darüber hinaus eröffnet die Erstellung individueller Nutzerprofile auch weitere Möglichkeiten, die zur Steigerung der allgemeinen Attraktivität der Website genutzt werden können. So kann auf Basis der in Kapitel 2.3 beschriebenen Auswertungen beispielsweise die Portalseite von c-lounge um eine 'What's hot?'-Empfehlung erweitert werden, welche diejenigen Communities vorstellt, die in der vorangegangenen Woche die größte Nutzeraktivität zu verzeichnen hatten, und eine 'What's near?'-Empfehlung im personalisierten Bereich eines Nutzers zeigt Communities an, die von Personen mit ähnlichen Profilen häufig genutzt werden.

Abschließend wird kurz skizziert, wie die Integration und Nutzung eines Profilingystems im Geschäftsmodell b) von c-lounge aussehen kann: in diesem Fall ist der kommerziell orientierte Betreiber nicht an den verschiedenen Interessen der Nutzer interessiert, sondern am konkreten Verhalten der Mitglieder seiner themenspezifischen Community, die sich meistens mit einer bestimmten Art von Produkten oder Dienstleistungen beschäftigt.

Dazu ist das Beobachten und Auswerten feingranularer Aktivitäten zu ermöglichen, insbesondere die Teilnahme an bestimmten Foren, z. B. bezüglich Support, Kritik und Innovationen, oder das Verhalten bei Votings.

Dies impliziert eine Auslegung der Profilingkomponente als mandantenfähiges System, das es jedem einzelnen Betreiber erlaubt, seine Community gemäß deren individueller Ausprägung und Zielsetzung zu analysieren, Konsumentengruppen zu segmentieren und individualisiert anzusprechen.

Idealerweise geschieht die hierzu notwendige Konfiguration und Nutzung des Systems, also das Anlegen und Ändern von Klassifikationen und Profilschemata, die semantische Klassifizierung von Content und das Durchführen von Analysen per Web-Client, wobei aufgrund der Komplexität dieser Bedienungsabläufe eine Realisierung als Java-Applet o. ä. gegenüber einer HTML-basierten Oberfläche vorzuziehen ist.

Es ist zu prüfen, ob die Implementierung der in Kapitel 4.2.3 beschriebenen Zusatzkomponenten unter Berücksichtigung dieser Gesichtspunkte erfolgen kann, um sie ggf. als universelle Client-Werkzeuge zu entwickeln, die den Anforderungen beider Einsatzzwecke gerecht werden.



Eine Erweiterung des Geschäftsmodells b) um leistungsfähige Profiling- und Analyse-Services trägt darüber hinaus zur Steigerung der Attraktivität dieser Dienstleistung für potenzielle Investoren bei.

## **5.2 Quinner**

In diesem Kapitel erfolgt die grobe Beschreibung einer möglichen Anwendung des in Kapitel 4 erstellten Referenzmodells für Profiling bei der B2C-Plattform der Quinner AG.

### **5.2.1 Geschäftsmodell von Quinner**

Die Quinner AG aus Griesheim ist ein Anbieter im Bereich der Online-Marktplätze für Versicherungen. Ziel der geschäftlichen Tätigkeit ist der Verkauf und die Vermittlung von Versicherungs- und Fondsprodukten per Internet. Dabei richtet sich Quinner zunächst an Endkunden und unterscheidet sich von Wettbewerbern durch die Tatsache, dass das Unternehmen als für die Beratungsleistung haftender Versicherungsmakler auftritt und nicht nur als Mittler zwischen Versicherungsgesellschaften und Kunden.

Die zentrale Grundlage dieser Geschäftsbeziehung bildet deshalb ein Serviceauftrag, der Quinner dazu bevollmächtigt, den Versicherungsbedarf des Kunden zu prüfen, ihn zu beraten, bestehende Versicherungen zu verwalten und nach dessen Weisung Versicherungen zu kündigen oder abzuschließen.

Die Beratung basiert hauptsächlich auf einem sogenannten 'Versicherungs-Check', der aus einer Reihe allgemeiner sowie von deren Beantwortung abhängiger spezieller Fragen zur Lebenssituation des Nutzers besteht. Aus diesen Angaben insbesondere zu Alter, Geschlecht, Familienstand, Beruf usw. wird eine Empfehlung abgeleitet, welche gängige Versicherungsprodukte aller Sparten, wie Sach-, Lebens- und Krankenversicherung als 'erforderlich', 'empfehlenswert' oder 'weniger wichtig' bzw. 'nicht notwendig' für den Nutzer einstuft.

Zu den mindestens als empfehlenswert eingestuften Versicherungen wird dem Nutzer bei Bedarf abschließend ein Preisvergleich angezeigt, der marktübliche Höchst- und Durchschnittspreise dem Preis einer entsprechenden Versicherung bei Quinner gegenüberstellt. Darüber hinaus hat der Interessent die Möglichkeit, für bereits vorhandene Versicherungen die bisher tatsächlich zu zahlenden Beiträge in einen sogenannten 'Sparrechner' einzugeben, um das Einsparungspotenzial durch die von Quinner vermittelten Produkte zu verdeutlichen.

Das Ergebnis dieser automatisierten Beratung folgt den Grundregeln, vorhandene, aber nicht notwendige Versicherungen ersatzlos zu kündigen, nicht vorhandene, aber erforderliche Versicherungen durch Quinner abzuschließen und sonstige sinnvolle und vorhandene Versicherungen durch preisgünstigere zu ersetzen, sofern Quinner diese offerieren kann. Kommt es daraufhin zum Abschluss eines Serviceauftrags, werden diesen Empfehlungen entsprechende Beauftragungen dem Vertragswerk beigelegt und kommen zur Ausführung, wobei der zum Kunden konvertierte Nutzer der Website einzelne Empfehlungen nach eigenem Ermessen ablehnen bzw. durch zusätzliche Beauftragungen ergänzen kann.

Im weiteren Verlauf der Geschäftsbeziehung will Quinner stets erster Ansprechpartner des Kunden sein, wobei die Meldung und Abwicklung von Schadensfällen eingeschlossen ist. Die Kommunikation zwischen beiden Parteien erfolgt dabei ausschließlich mittels elektronischer Medien oder postalisch, wobei die Nutzung der Internetplattform priorisiert

wird und ergänzende Beratung durch Servicemitarbeiter in einem Call Center zur Verfügung steht. Um den heutzutage hohen Ansprüchen an die Convenience einer solchen Kommunikation gerecht zu werden, bietet Quinner u. a. einen sogenannten 'Call-Back'-Button auf allen erklärungsbedürftigen Webseiten an, durch dessen Nutzung Interessenten und Kunden einen sofortigen oder späteren Rückruf durch das Call Center veranlassen können. Dabei ist auch Co-Browsing möglich, d. h. das gleichzeitige Anzeigen einer geöffneten Webseite beim Nutzer und Servicemitarbeiter, wodurch eine interaktive Hilfestellung, insbesondere beim Ausfüllen von Online-Formularen, geboten werden kann.

Durch diesen gänzlichen Verzicht auf Außendienstmitarbeiter erhofft sich Quinner die Kostenvorteile einer Direktversicherung, die neben der Aushandlung günstiger Tarife mit den Versicherungsanbietern zu Einsparungen für die Kunden führen sollen.

Für diese ist die Maklerdienstleistung kostenlos, da sich Quinner durch die von den Versicherungsgesellschaften ausgeschütteten Vermittler- und Bestandsprovisionen finanziert.

Darüber hinaus ist zu einem späteren Zeitpunkt der Verkauf von Fondsprodukten geplant, die Kunden zur Anlage der im Versicherungsbereich eingesparten Beiträge und damit zum Vermögensaufbau nutzen sollen.

### 5.2.2 Profiling bei Quinner

Im Hinblick auf die Nutzung der Website [www.quinner.de](http://www.quinner.de) muss grundsätzlich zwischen Interessenten und bestehenden Kunden der Quinner AG unterschieden werden. Da letztgenannten ein hohes Maß an Serviceleistungen geboten wird und eine Verwaltung umfangreicher Vertrags- und Abwicklungsdaten notwendig ist, bilden Website und Call Center nur das Frontend einer umfassenden Systemlandschaft, die im nächsten Abschnitt skizziert wird.

Da bei Quinner bereits alle Austauschprozesse zwischen Anbieter und Kunde erfasst und historisiert werden, um zu jedem Zeitpunkt eine vollständige Sicht auf den Kunden zu haben und notwendige Geschäftsprozesse elektronisch unterstützt initiieren und ablaufen zu lassen, ist ein zusätzliches feingranulares Profiling bestehender Kunden auf der Website nicht sinnvoll.

Durch Architektur und Realisierung der DV-Infrastruktur wird Quinner auch den in Kapitel 2.1.4 aufgestellten Anforderungen einer integrativen Handhabung verschiedener Kommunikationskanäle gerecht, insbesondere die systemtechnische Einbindung von CTI-Anwendungen und einem Dokumentenmanagementsystem trägt dazu bei.

Von besonderem Interesse ist dagegen das Verhalten der Interessenten, die sich erstmalig und zunächst anonym auf die Website begeben, da das Internet die einzige Anlaufstelle für potenzielle Neukunden von Quinner darstellt. Einer einfachen oder zumindest leicht verständlichen Nutzerführung sowie frühen und starken Anreizen zur Fortsetzung der Interaktion über den Versicherungs-Check hinaus bis zur Erteilung eines Serviceauftrags kommt damit eine strategische Bedeutung zu, die heutzutage so wichtige Konversion von Interessenten zu Kunden zu fördern, welche vielen Online-Anbietern von Produkten oder Dienstleistungen nach wie vor schwer fällt (vgl. Kapitel 1.1).

Die durch eine Realisierung des Referenzmodells möglichen Profilingmaßnahmen erlauben hierzu ein exaktes Beobachten einzelner Nutzer sowie die Ermittlung typischer Verhaltensmuster ihrer Gesamtheit. Damit könnten kritische Stellen im durch den Versicherungs-Check vorgegebenen Workflow identifiziert und vorhandene Schwächen ggf. beseitigt werden. Insbesondere ist ein Aufzeigen typischer Abbruchstellen möglich, sofern vorhanden, und evtl. gemeinsame Merkmale nicht wiederkehrender Nutzer können entdeckt werden.

Aus derartigen Erkenntnissen lassen sich ebenfalls Modifikationen der Werbemaßnahmen von Quinner ableiten und auch eine Nutzung als Controlling-Instrument der Marketingstrategie ist denkbar.

Profiling im Bereich des Sparrechners kann zur Feststellung der den Nutzern tatsächlich gebotenen Einsparungsmöglichkeiten genutzt werden, um diese werbewirksam zu kommunizieren.

Ein weiterer Aspekt, zu dessen Betrachtung Profiling als Methode sinnvoll erscheint, ist die Akzeptanz der von Quinner ausgesprochenen Empfehlungen bezüglich erforderlicher, empfehlenswerter und weniger wichtiger Versicherungen bzw. das Maß der von den Nutzern praktizierten Abweichung hiervon. So könnte sich herausstellen, dass bestimmte als sinnvoll eingestufte Versicherungen häufig nicht erwünscht sind oder der Abschluss einiger 'nicht notwendiger' Versicherungen dennoch beauftragt wird.

Die in Kapitel 2.3.3 vorgestellten Analysen auf Profildaten können potenzielle Zusammenhänge zwischen diesem Verhalten und individuellen Merkmalen der Nutzer aufzeigen, was zur Verfeinerung der Empfehlungsregeln oder Erschließung von *Cross-* und *Up-Selling*-Möglichkeiten genutzt werden kann.

Eine Anwendung des in Abschnitt 4.2.3.3 beschriebenen 'Analyzer'-Werkzeugs sollte jedoch möglichst direkt auf den bei Quinner verfügbaren Daten erfolgen, da eine Nachbildung dieser komplexen Informationen in einem parallel arbeitenden Profilingssystem als zu aufwändig einzustufen ist.

Aus den oben genannten Gründen befassen sich die nachfolgenden Beschreibungen der Architektur von Quinner und eines möglichen Einsatzes des Profilingssystems nur noch mit den für die Beobachtung der anonymen Websitebesucher relevanten Aspekten.

### 5.2.3 Beschreibung der Webanwendung von Quinner

Der komplexe system- und softwaretechnische Aufbau der DV-Infrastruktur bei Quinner begründet sich aus der Anforderung, einer großen Anzahl potenzieller Kunden die Kommunikation mit dem Unternehmen per WWW, e-Mail, Fax, Post und Telefon zu ermöglichen und die dadurch initiierten Geschäftsprozesse z. T. automatisiert sowie mithilfe verschiedener Teams und durch ein Workflowsystem gesteuert ablaufen zu lassen. Als weitere Beteiligte müssen darüber hinaus Versicherungsgesellschaften, Banken, Anspruchsteller, Schadenregulierer usw. berücksichtigt werden, deren Datenaustausch mit Quinner teilweise ebenfalls auf elektronischem Wege geschieht. Zur Minimierung des Papieraufkommens wird zur Verarbeitung eingehender Post und Faxnachrichten ein Dokumentenmanagementsystem eingesetzt, das auf der Verteilung digitalisierter Schriftstücke basiert.

Die Website von Quinner, die als erste Anlaufstelle für Interessenten fungiert, wird durch ein System aus Web- und Applikationsservern realisiert, die sich einer Datenbank bedienen und mit einem CTI-, Workflow- und Administrationssystem interagieren.

Innerhalb des Applikationsservers der Firma Inprise kommt Java in Form von JSP, Servlets und EJB-Komponenten (Vers. 1.1) zum Einsatz, wobei HTML- und JSP-Seiten die Präsentation von Content übernehmen, Servlets die im Rahmen der HTTP-Kommunikation aufwändigeren Aufgaben wie Nutzerregistrierung, Login, Fehlerbehandlung und Call-Back-Funktion realisieren und EJBs für die Erbringung von Kernfunktionalitäten zuständig sind, beispielsweise die Anbindung an andere Systeme oder die Rechenlogik des oben erwähnten Sparrechners.

Dieser interne Aufbau bleibt Nutzern der Website jedoch verborgen; die zunächst anonymen Interessenten absolvieren im Wesentlichen den Versicherungs-Check, der wie folgt abläuft:

1. Zunächst sind drei stets gleich bleibende Webseiten mit allgemeineren Fragen zu durchlaufen, die sich auf Geschlecht, Alter, Familienstand, Berufstätigkeit, Einkommen sowie Besitz von Häusern, Wohnungen, Kfz und Haustieren beziehen.
2. Danach erfolgt die Anzeige der von Quinner empfohlenen Einstufung der verschiedenen Versicherungssparten als 'erforderlich', 'empfehlenswert' und 'weniger wichtig' bzw. 'nicht notwendig' (vgl. Abbildung 36).
3. Anschließend folgt eine variable Anzahl weiterer HTML-Formulare, deren Inhalt von den in den vorangegangenen Schritten getätigten Angaben abhängt. So führt das Vorhandensein eines Pkw und die Einwilligung in eine diesbezügliche Versicherungsberatung zu Fragen zum Fahrzeugtyp, der jährlichen Fahrleistung usw. Analog verhält es sich mit näheren Angaben zu weiteren Versicherungen, wie beispielsweise Privat- und Tierhalterhaftpflicht, Rechtsschutz-, Unfall- und Hausratversicherung usw.
4. Die vollständige Erfassung der notwendigen Angaben führt zur Auswahl von Versicherungsprodukten mit einem günstigem Preis-Leistungs-Verhältnis aus einer Produktdatenbank, deren Beiträge daraufhin im Vergleich zu marktüblichen Höchst- und Durchschnittsbeiträgen angezeigt werden. Es besteht die Möglichkeit, den bereits beschriebenen Sparrechner zu verwenden oder Versicherungen für weitere Kfz oder Gebäude zu erfragen.

Auf allen Seiten des Versicherungs-Checks ist das Zwischenspeichern der gemachten Angaben möglich, wozu die Registrierung unter einem beliebigen Nutzernamen notwendig wird. Bereits registrierte Nutzer können sich jederzeit einloggen und auf bereits vorhandene Daten zurückgreifen.

Nach der Überprüfung des Versicherungsbedarfs kann der Nutzer mit der Erteilung eines Serviceauftrags fortfahren, wozu seine Deanonymisierung erforderlich wird, nähere Angaben über zu kündigende Versicherungen erfragt werden sowie das als PDF-Datei individuell generierte Vertragswerk ausgedruckt, unterschrieben und postalisch versendet werden muss.

Die weitere Kommunikation von als Kunden gewonnenen Nutzern mit Quinner erfolgt im Web mittels eines personalisierten Bereichs, der eine Anmeldung mit Nutzernamen und Passwort voraussetzt.

Dort können die individuellen Stamm- und Vertragsdaten eingesehen sowie Änderungen und Schadensmeldungen veranlasst werden.

The screenshot shows the Quinner.de website in a Microsoft Internet Explorer browser. The page title is 'Quinner.de - Microsoft Internet Explorer von Lycos Bertelsmann'. The address bar shows the URL: <https://www2.quinner.de/quinner.jsp/NSID-www2.quinner.de-973%3A3b38be8b%3Aa59b3ce277dd1d5?fp=34d56>. The website has a navigation menu with links: Home, My Quinner, Angebot, Unternehmen, Presse, Kontakt, Info, and AGB. The main content area is titled 'Die Quinner-Empfehlung' and includes a sub-header 'Check-Resultat' and 'Seite 1/1'. Below this is a table with three columns: 'Empfehlung', 'Versicherung', and 'interessiert mich'. The table is divided into three sections: 'erforderliche Versicherungen', 'empfehlenswerte Versicherungen', and 'weniger wichtige oder aufgrund Ihres Checks nicht notwendige Versicherungen'. Each section lists insurance types with checkboxes for interest.

Empfehlung	Versicherung	interessiert mich
erforderliche Versicherungen	<a href="#">KFZ</a>	<input checked="" type="checkbox"/>
	<a href="#">Privathaftpflicht</a>	<input checked="" type="checkbox"/>
	<a href="#">Auslandsreisekranken</a>	<input checked="" type="checkbox"/>
	<a href="#">Berufsunfähigkeit</a>	<input checked="" type="checkbox"/>
empfehlenswerte Versicherungen	<a href="#">Unfall</a>	<input checked="" type="checkbox"/>
	<a href="#">Rechtsschutz</a>	<input checked="" type="checkbox"/>
	<a href="#">Hausrat</a>	<input checked="" type="checkbox"/>
	<a href="#">Krankenzusatz</a>	<input checked="" type="checkbox"/>
	<a href="#">Risikoleben</a>	<input checked="" type="checkbox"/>
weniger wichtige oder aufgrund Ihres Checks nicht notwendige Versicherungen	<a href="#">Tierhalterhaftpflicht</a>	<input type="checkbox"/>
	<a href="#">Wohngebäude</a>	<input type="checkbox"/>
	<a href="#">Kinderunfall</a>	<input type="checkbox"/>
	<a href="#">Gewässerschadenhaftpflicht</a>	<input type="checkbox"/>

Abbildung 36: Screenshot Quinner-Empfehlung

#### 5.2.4 Einsatz des Referenzmodells für Profiling bei Quinner

Aus den in Kapitel 5.2.2 diskutierten Gründen bleibt die Anwendung des Referenzmodells für Profiling auf den allen Interessenten zugänglichen Bereich der Website von Quinner beschränkt, was den Versicherungs-Check bis hin zum Abschluss eines Serviceauftrags einschließt.

Da neben diesem exakt definierten, nahezu sequentiellen Workflow nur noch wenige, fast unveränderliche und sehr allgemeine Webseiten abrufbar sind ('Home', 'Unternehmen', 'Presse', 'Kontakt', 'Info', 'AGB'), kann zu deren Klassifizierung der jeweilige Seitentitel verwendet werden. Darüber hinaus sind natürlich alle Angaben von Interesse, die Nutzer im Laufe des Versicherungs-Checks machen, wobei neben den oben bereits aufgezählten

Daten hinsichtlich Alter, Beruf usw. insbesondere das Interesse an den verschiedenen Arten von Versicherungsprodukten in das Profil aufzunehmen ist.

Die Erstellung einer Klassifikation und eines Profilschemas kann somit darauf reduziert werden, mittels numerischer Attribute die Anzahl der Pageviews aller vorkommenden Seiten zu zählen und die im Rahmen des Versicherungs-Checks ermittelbaren Informationen im Profil zu hinterlegen. Die potenziell zu vermittelnden Versicherungsprodukte können flach oder hierarchisch in das Profilschema integriert werden, wobei pro Produkt mindestens erfasst werden muss, ob der Nutzer Interesse daran hat und ob er bereits eine Versicherung dieser Art besitzt. Ein in den Sparrechner eingegebener Beitrag kann ebenfalls gespeichert werden, um später auf das Maß der Ersparnis zugreifen zu können oder Anhaltspunkte für durchschnittliche Versicherungsbeiträge zu erhalten.

Eine hierarchische Strukturierung der von Quinner vermittelten Produkte ist dann vorzuziehen, wenn Fondsprodukte angeboten werden, damit zwischen Versicherungen und Finanzanlagen differenziert werden kann.

Für die erforderliche Abbildung des Workflows des Versicherungs-Checks ist der in Abschnitt 4.2.1.4 modellierte Mechanismus geeignet, welcher mithilfe eines Automaten den Fortschritt einer geschäftlichen Transaktion überwacht. Hierdurch wird die genaue Beobachtung eines solchen Ablaufs möglich, wobei eine gesonderte Abspeicherung von abgeschlossenen und abgebrochenen Transaktionen dazu beiträgt, mögliche Gründe hierfür zu ermitteln.

Eine Implementierung des Referenzmodells für Profiling sollte für den Einsatz bei Quinner in Java gemäß EJB-Standard erfolgen, da dort bereits ein EJB-fähiger Applikationsserver eingesetzt wird. Auf diese Weise kann eine gute Integration in die bestehende Webanwendung sichergestellt werden; die Installation zusätzlicher Software ist nicht notwendig und der Ablauf von Quinner-Applikation und Profilingkomponente im selben Serversystem ermöglicht ein gemeinsames Session-Management und den einfachen Zugriff auf die Schnittstellen zum Lesen und Schreiben von Profilinginformationen.

Die Ausnutzung der vom Applikationsserver bereitgestellten Dienste (vgl. Kapitel 3.2.1.3) vereinfacht die Realisierung des Profiling auf Basis des Referenzmodells ebenfalls, da u. a. die persistente Speicherung von Objekten erleichtert wird. So kann der Einsatz von EntityBeans mit 'container-managed persistence' dazu beitragen, den Entwickler von Details der Datenhaltung zu befreien [Roma99].

### 5.2.5 Bewertung und Ausblick

Die hier grob skizzierte Anwendung des Referenzmodells für Profiling bei Quinner macht deutlich, dass Internetanwendungen, die bereits stark personalisierte Abläufe von Geschäftsprozessen ermöglichen und in hohem Maße individuelle, teils charakterisierende Nutzerdaten enthalten, nur eingeschränkt durch ein zusätzliches Profilingssystem sinnvoll ergänzt werden können. Es ergibt sich insbesondere das Problem einer doppelten Datenhaltung, die grundsätzlich zu vermeiden ist, sowie ein hoher Integrationsaufwand, falls Anbindungen an die CTI-, Dokumentenmanagement- und Workflow-Software geschaffen werden müssten.

Das Beispiel Quinner zeigt jedoch auch, dass der Einsatz einer Profilingkomponente für einen Teilbereich einer Webapplikation nutzbringend sein kann, in diesem Fall zur Bildung von Profilen noch anonymer Interessenten, welche durch diesen Status nicht der vom System unterstützten Bildung einer Kundenhistorie unterliegen.

Die exakte Aufzeichnung des Nutzerverhaltens in Verbindung mit den im Rahmen der Online-Befragung gewonnenen Daten kann maßgeblich zur Erreichung der in Abschnitt 5.2.2 zur Motivation genannten Ziele beitragen:

- Identifikation kritischer Stellen im Versicherungs-Check, an denen überproportional häufig ein Abbruch des Workflows geschieht
- Analyse, ob es zudem charakteristische Merkmale der Nutzer gibt, die ihren Versicherungs-Check abbrechen oder zwischengespeicherte Sitzungen nicht wieder fortsetzen
- Analyse der möglichen Abweichung der Nutzer gegenüber den von Quinner ausgesprochenen Empfehlungen hinsichtlich des Versicherungsbedarfs

Die Ergebnisse dieser Untersuchungen können zu einer Optimierung des vorgegebenen Workflows oder der erklärenden Texte auf den Webseiten beitragen. Darüber hinaus sind Erkenntnisse für die Modifikation von Werbemaßnahmen für Quinner erreichbar.

Möglicherweise wird festgestellt, dass sich unerwartet viele Nutzer mit eher technisch-handwerklichen Berufen (Frage zu Beginn des Versicherungs-Checks) für das Dienstleistungsangebot von Quinner interessieren, was bei der Media-Selektion zur Schaltung von Print-Werbung ausgenutzt werden kann. Weiterhin ist denkbar, dass Nutzer, die den Sparrechner ausprobieren, mit einer signifikant höheren Wahrscheinlichkeit zu Kunden konvertieren, weshalb auf dessen Verwendung besonders hingewiesen werden sollte. Die Erfassung der in den Sparrechner eingegebenen Werte kann zur Anzeige des durchschnittlichen oder maximalen Einsparungspotenzials auf der Startseite von [www.quinner.de](http://www.quinner.de) genutzt werden, beispielsweise in der Form: 'Die durchschnittliche Einsparung unserer Neukunden der letzten Woche betrug 237,- DM. Die zehn größten Sparer freuen sich sogar über rund 813,- DM!'.

Es ist jedoch zu prüfen, ob der Aufwand der Integration eines Profilingystems auf Basis des erstellten Referenzmodell den einer proprietären Weiterentwicklung der Quinner-Anwendung übersteigt, mithilfe derer der gleiche Nutzen angestrebt werden kann.



## **6 Fazit und Ausblick**

Abschließend sollen die erarbeiteten Anforderungen an ein Referenzmodell für Profiling sowie dessen grober Entwurf hinsichtlich einer zu erfolgenden Implementierung kritisch betrachtet werden. Die Bewertung der Einsetzbarkeit und des dadurch erbrachten Nutzens führt zu einer Idee für eine mögliche Realisierung. Darüber hinaus werden weiterführende Aufgaben aufgezeigt.

### **6.1 Anforderungen und Entwurf**

Obwohl in Kapitel 2 ein weites Feld von Anforderungen an Profiling aufgespannt wird, die es im Rahmen einer effektiven Gewinnung, Verwaltung und Nutzbarmachung von Profildaten zu beachten gilt, können sich diese im Einzelfall als nicht ausreichend erweisen. Je nach Zielsetzung des Einsatzes von Profilingmaßnahmen ist die Forderung weiterer Eigenschaften denkbar.

Für die Implementierung eines auf diesem Referenzmodell basierenden Systems ist zudem eine Verfeinerung sowohl der Anforderungen als auch des Entwurfs erforderlich, welche jedoch außerhalb des Fokus dieser Arbeit lag.

Insbesondere die Ausarbeitung der nur grob beschriebenen Anforderungen hin zu einem detaillierten Anforderungsdokument bringt weitere, spezifischere Maßgaben hervor, deren hohe Anzahl und unterschiedliche Ziele zu Inkonsistenzen führen können. Beispielsweise wirkt die Forderung nach einer möglichst umfassenden und damit komplexen Profildatengewinnung i. Allg. dem Wunsch nach Performanz eines solchen Systems inhärent entgegen.

Somit besteht eine weiterführende Aufgabe darin, eventuell bestehende Lücken innerhalb der Anforderungen zu schließen, diese zu verfeinern sowie Widersprüche zu identifizieren und durch die Vergabe und Bewertung von Prioritäten aufzulösen.

Die auf dem abgeschlossenen Anforderungssystem aufbauende Spezifikation und der Feinentwurf werden im gleichen Maße über die in Kapitel 4 auf wesentliche Aspekte beschränkte Modellierung hinausgehen, um diese zu einem vollständigen Softwarefeinentwurf zu erweitern. Dabei sind vor allem zahlreiche für eine Implementierung notwendige Hilfsdatenstrukturen zu entwickeln, beispielsweise für eine effiziente Umsetzung der Semantik-Tabelle oder einzelne Personalisierungsregeln; aber auch die Auswahl und Entwicklung grundlegender Algorithmen ist notwendig, z. B. für ein Pattern Matching sowie Operationen auf und Analysen von Profilen, um letztendlich alle zuvor gestellten Anforderungen zu berücksichtigen.

Schließlich kann auch der Einsatz eines auf diesem Referenzmodell basierenden Systems in der Praxis weitere Anforderungen aufzeigen, die sich aufgrund von neuartigen Geschäftsmodellen, zu integrierenden Fremdsystemen oder Fortschritten bei zugrunde liegenden Technologien ergeben.

### **6.2 Einsetzbarkeit und Nutzen**

Die in Kapitel 5 skizzierte Anwendung des Referenzmodells für Profiling bei c-lounge und Quinner belegt dessen prinzipielle Einsetzbarkeit und zeigt darüber hinaus den jeweils erreichbaren Nutzen auf.

Der modulare Aufbau und der Betrieb eines solchen Profilingssystems als separate Komponente erleichtern in beiden Fällen die systemtechnische Integration in die bestehende Webanwendung, verursachen jedoch den bereits in Abschnitt 4.2.1.1 kritisierten Aufwand der zumindest teilweisen Nachbildung der Applikationslogik in der Profilingkomponente.

Das Anwendungsbeispiel c-lounge macht deutlich, dass dieser Aufwand vergleichsweise gering bleiben kann, während ein für die Erreichung der wirtschaftlichen Ziele der Website essentieller Nutzen erbracht wird: die bei c-lounge bisher nicht umgesetzte Generierung und Haltung charakterisierender Profile, die für die Einblendung personalisierter Werbung vonnöten ist, kann vollständig von einem System erbracht werden, das auf dem erstellten Referenzmodell basiert. Da nur wenige Eigenschaften der Nutzer abzubilden sind, die durch den strukturierten Aufbau der Webanwendung zudem leicht durch das Abspalten aller Client-Requests erfasst werden können, hält sich neben der Inbetriebnahme auch der Aufwand für die gelegentliche Anpassung des Profilingssystems in engen Grenzen.

Bei Quinner ist der Einsatz des Referenzmodells jedoch nur eingeschränkt sinnvoll, da die dort vorhandene Webanwendung und die damit verbundenen Informationssysteme bereits in hohem Maße Kundendaten enthalten, die alle Interaktionen mit dem Unternehmen protokollieren und die Kunden charakterisierend beschreiben. Die zusätzliche Verwendung einer Profilingkomponente würde zu einer teilweise redundanten Datenhaltung und hohem Integrationsaufwand führen, da zur Beibehaltung einer ganzheitlichen Sicht auf die Kunden Schnittstellen zu den CTI- und Dokumentenmanagementsystemen geschaffen werden müssten.

Empfehlenswert ist jedoch der Einsatz von Profilingmaßnahmen in der Anbahnungsphase einer Kundenbeziehung mit Quinner sowie die Durchführung der dadurch möglich werdenden Analysen des tatsächlichen Beauftragungsverhaltens für die offerierten Versicherungsprodukte. Hierdurch kann der initiale und wichtigste Geschäftsprozess im Lebenszyklus eines Quinner-Kunden durch Beobachtung dessen Ablaufs detailliert bewertet werden, was für eine Prozessoptimierung nützlich ist. Darüber hinaus können durch Abweichungen der Nutzer von den ausgesprochenen Empfehlungen ggf. Cross- und Up-Selling-Potenziale erschlossen werden.

Es bleibt zu prüfen, ob diese in jedem Fall sinnvollen Maßnahmen leichter durch den Einsatz eines auf dem erstellten Referenzmodell basierenden Systems zu realisieren sind oder eine Weiterentwicklung der bestehenden Anwendung weniger Aufwand verursacht. Dabei muss sich auch die Umsetzung der letztgenannten Möglichkeit an den Anforderungen an Profiling orientieren und profitiert somit von deren Erarbeitung in Kapitel 2.

### 6.3 Vision einer Realisierung

Zusammenfassend wird festgestellt, dass das erstellte Referenzmodell für Profiling bei e-Business-Websites als Basis der Realisierung eines Softwaresystems dienen kann, dessen Einsatz einen deutlich erkennbaren Nutzen für die Betreiber erbringt, an dem auch Konsumenten partizipieren können. Daraus erwächst eine Bedeutsamkeit, die mit der in nächster Zeit erwarteten steigenden Verbreitung von auf Profiling basierenden Maßnahmen weiter zunimmt.

Deshalb wird angeregt, das hier erstellte Modell nach der notwendigen Verfeinerung von Anforderungen und Entwurf als Softwarekomponente zu realisieren, die einem Standard-Komponentenmodell entspricht und als Zusatz- oder Basiskomponente einer Web-basierten, kundenfokussierten Anwendung dienen kann. In Verbindung mit den im Kapitel 4.2.3 skizzierten Werkzeugen kann ein Softwaresystem entstehen, das sogar eher einem eigenständigen Produkt als einer *COTS-Komponente* entspricht. Hierzu ist jedoch eine qualitativ hochwertige Realisierung notwendig, welche im Idealfall eine frühe Integration mit existierenden Webanwendungen prototypisch versucht, damit die dadurch erzielte Reife der Software den Produktcharakter unterstützt.

Als zu verwendendes Komponentenmodell empfiehlt sich der EJB-Standard gemäß [MaHa99], der eine strukturierte, grundlegend objektorientierte Implementierung in Java ermöglicht und von zahlreichen Applikationsservern unterstützt wird. Derartige technische Lösungen besitzen bereits eine hohe Verbreitung im Bereich der Internet-basierten Geschäftsprozessabwicklung und bieten darüber hinaus, z. B. mittels CORBA, Schnittstellen zu anderen Standardarchitekturen.

Auf diese Weise wird auch ein einheitlicher und insgesamt vereinfachter Einsatz eines solchen Systems bei c-lounge und Quinner möglich, sofern der momentan bei c-lounge verwendete Applikationsserver auf eine EJB-fähige Nachfolgeversion aufgerüstet wird.

Eine weitere Voraussetzung für die Vermarktung eines derartigen Produkts ist die Verfügbarkeit von Varianten für möglichst viele der verbreiteten Applikationsserver, wobei die Profilingkomponente zwecks leichter Integration und Lastverteilung auch allein auf einem dedizierten Server lauffähig sein muss.

Der zuvor dargestellte Nutzen ergibt sich jedoch nicht nur bei der Erweiterung einer bestehenden Webanwendung um das Profilingssystem als zusätzliche Komponente; die Verwendung als Basis einer neu zu erstellenden Webapplikation ist ebenfalls möglich und verspricht Vorteile, wenn dabei die Eigenschaften der Nutzer einer Website ins Zentrum der Applikation gerückt werden sollen, um sämtliche Geschäftsprozesse daran auszurichten.

Dies wird deutlich, wenn man sich die Entwicklung der Quinner-Systemlandschaft bei Vorhandensein einer flexiblen und ausgereiften Basiskomponente für das Profiling vorstellt: alle erforderlichen Stammdaten könnten in das Profilschema integriert werden, um die Bestandsführung zu vereinfachen, und die Zusammenführung der Interaktionsprotokolle zu einer Kundenhistorie würde im Profilingssystem geschehen (vgl. Klasse *Interaction* in Abbildung 11). Durch die in Kapitel 4 modellierten Schnittstellen wäre ein Zugriff auf

Kundendaten für die Mitarbeiter des Call Centers leicht möglich und die für Quinner besonders interessanten Analysen der Profile würden tatsächliche Kunden und anonyme Interessenten einschließen.

Als weiterführende Aufgabe kann somit neben der Verfeinerung von Anforderungen und Entwurf die Implementierung und Evaluierung des erstellten Referenzmodells für Profiling bei e-Business-Websites in der Praxis gelten, da die in Kapitel 5 stattgefundenen Instanzierungen auf Anwendungsbeispiele den Status eines Gedankenexperiments nicht verlassen.

**ABBILDUNGSVERZEICHNIS**

Abbildung 1: Prognose der globalen Umsatzentwicklung B2C und B2B [ECIN01]	9
Abbildung 2: Beispiel einer Kategorisierung	31
Abbildung 3: Visualisierung von Navigationsverhalten	39
Abbildung 4: Vektorendarstellung von Profilattributen	40
Abbildung 5: Beispiel einer grafischen Darstellung von Abhängigkeiten in Profildaten	41
Abbildung 6: Lastverteilung zwischen Webservern	55
Abbildung 7: Anwendungsfalldiagramm Profiling	72
Abbildung 8: Klassendiagramm <code>User</code>	74
Abbildung 9: Klassendiagramm <code>Request</code>	75
Abbildung 10: Klassendiagramm <code>Session - Transaction</code>	76
Abbildung 11: Klassendiagramm <code>Profile</code>	77
Abbildung 12: Verteilungsdiagramm Profiling	79
Abbildung 13: Klassendiagramm Sessionmanagement	81
Abbildung 14: Klassendiagramm <code>SemanticsTable</code> und <code>TransactionTable</code>	83
Abbildung 15: Klassendiagramm <code>Transaction</code>	85
Abbildung 16: Zustandsdiagramm Online-Bestellung	86
Abbildung 17: Kollaborationsdiagramm Request-Abspaltung	87
Abbildung 18: Kollaborationsdiagramm Request-Abspaltung mit Personalisierung	88
Abbildung 19: Sequenzdiagramm Sessionstart	89
Abbildung 20: Sequenzdiagramm Profilerweiterung	91
Abbildung 21: Sequenzdiagramm Transaktion	91
Abbildung 22: Sequenzdiagramm Sessionende	93
Abbildung 23: Klassendiagramm <code>Composite</code>	95
Abbildung 24: Klassendiagramm Profilinstanz	96
Abbildung 25: Klassendiagramm <code>SchemaBuilder</code>	97
Abbildung 26: Komponenten-/Verteilungsdiagramm Gesamtsystem	103
Abbildung 27: Screenshot c-lounge Portal	108
Abbildung 28: Sitestruktur c-lounge Community-Mitglied	109
Abbildung 29: Sitestruktur c-lounge Community-Betreiber	110
Abbildung 30: Komponentendiagramm c-lounge [Hand01]	112
Abbildung 31: Komponenten-/Verteilungsdiagramm c-lounge	113
Abbildung 32: Kategorisierung der c-lounge-Communities	115
Abbildung 33: Profilschema c-lounge	116
Abbildung 34: Komponenten-/Verteilungsdiagramm c-lounge mit Profiling	117
Abbildung 35: Aktivitätsdiagramm c-lounge-Profiling	119
Abbildung 36: Screenshot Quinner-Empfehlung	128

## **TABELLENVERZEICHNIS**

Tabelle 1: Markt- und Transaktionsbereiche des e-Commerce [HaSe99]	9
Tabelle 2: DCMES-Elemente für Metainformationen – Version 1.1 [DCMI01b]	63
Tabelle 3: Hauptklassen der Dewey Decimal Classification [DDC01a]	64

**ABKÜRZUNGSVERZEICHNIS**

<b>ACID</b>	Atomicity, Consistency, Isolation, Duration
<b>ACM</b>	Association for Computing Machinery
<b>AMCIS</b>	Americas Conference on Information Systems
<b>API</b>	Application Programming Interface
<b>APPEL</b>	A P3P Preference Exchange Language
<b>ASP</b>	Active Server Pages
<b>B2B</b>	Business-to-Business
<b>B2C</b>	Business-to-Consumer
<b>CGI</b>	Common Gateway Interface
<b>CMS</b>	Content-Management-System
<b>CORBA</b>	Common Object Request Broker Architecture
<b>COTS</b>	Commercial off the Shelf
<b>CPEXchange</b>	Customer Profile Exchange
<b>CRM</b>	Customer Relationship Management
<b>CTI</b>	Computer Telephony Integration
<b>DC</b>	Dublin Core
<b>DCMES</b>	Dublin Core Metadata Element Set
<b>DCMI</b>	Dublin Core Metadata Initiative
<b>DDL</b>	Data Definition Language
<b>EDI</b>	Electronic Data Interchange
<b>FTP</b>	File Transfer Protocol
<b>HTML</b>	Hypertext Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>ICSE</b>	International Conference on Software Engineering
<b>IETF</b>	Internet Engineering Task Force
<b>IP</b>	Internet Protocol
<b>ISAPI</b>	Internet Server Application Programming Interface
<b>J2EE</b>	Java 2 Platform Enterprise Edition
<b>JSP</b>	JavaServer Pages
<b>LDAP</b>	Light Weight Directory Access Protocol
<b>MIME</b>	Multipurpose Internet Mail Extension
<b>NSAPI</b>	Netscape Server Application Programming Interface
<b>OPS</b>	Open Profiling Standard
<b>P3P</b>	Platform for Privacy Preferences
<b>PIN</b>	Persönliche Identifikationsnummer
<b>POP3</b>	Post Office Protocol, Version 3
<b>RFC</b>	Request for Comment
<b>RMI</b>	Remote Method Invocation
<b>SGML</b>	Standard Generalized Markup Language
<b>SIGecom</b>	Special Interest Group on Electronic Commerce
<b>SIGKDD</b>	Special Interest Group on Knowledge Discovery in Data
<b>SMS</b>	Short Message Service
<b>SMTP</b>	Simple Mail Transfer Protocol
<b>SQL</b>	Structured Query Language
<b>TAN</b>	Transaktionsnummer
<b>TCP</b>	Transmission Control Protocol

<b>TCP/IP</b>	Transmission Control Protocol/Internet Protocol (s. Internet Protocol)
<b>UDP</b>	User Datagram Protocol
<b>UML</b>	Unified Modeling Language
<b>URI</b>	Uniform Resource Identifier
<b>URL</b>	Uniform Resource Locator
<b>W3C</b>	World Wide Web Consortium
<b>WAP</b>	Wireless Application Protocol
<b>WWW</b>	World Wide Web
<b>XML</b>	Extensible Markup Language



## GLOSSAR

### Adserver

Webserver, der dem Bereitstellen von Werbebannern im Internet dient. Zu dieser speziellen Aufgabe gehört die quantitative Erfassung der Einblendungen sowie die Rotation der Banner-Motive.

nach [Stol99]

### Clickstream

Eine Abfolge von einzeln und bewusst ausgeführten HTML-Seitenabrufen (Clicks) durch einen Nutzer.

### Click-Through-Rate

Anzahl der Clicks auf ein Werbebanner im Verhältnis zur Gesamtzahl seiner Einblendungen.

### Commercial off the Shelf

Standard-Software oder –System, das einfach beschafft und unverändert eingesetzt werden kann.

### Common Gateway Interface

Schnittstelle von Webservern, die das Weiterleiten eines HTTP-Requests an ein ausführbares Programm auf dem Server erlaubt, welches die Client-Daten verarbeiten kann und für das Senden einer Antwort zuständig ist.

nach [OHE99]

### Common Object Request Broker Architecture

Standardisierte Architektur zur Verteilung und Nutzung verschiedenartiger Softwarekomponenten in Server-Umgebungen.

### Content

Redaktioneller Inhalt eines Online-Angebots.

### Content-Management-System

Softwaresystem zur Verwaltung und Pflege von redaktionellen Inhalten bei Websites.

### COTS-Komponente

Standardkonforme Softwarekomponente, die von einem Hersteller zur Integration in Eigenentwicklungen erworben werden kann, um Entwicklungsaufwand zu sparen (vgl. COTS).

### Cross-Selling

Strategie beim Verkauf, bei der Kunden der Erwerb weiterer Produkte nahegelegt wird, die mit dem gekauften ggf. in Verbindung stehen, z. B. indem sie es ergänzen (vgl. Up-Selling).

### Customer Relationship Management

Marketing-Sichtweise, die Aufbau und Pflege einer Kundenbeziehung in den Vordergrund stellt.

nach [Meff98]

### Data Definition Language

Untermenge der Structured Query Language (s. u.), welche die Definition von Daten und deren Zusammenhängen in Datenbanken erlaubt.

### Diversifikation

Die "Ausrichtung der Unternehmensaktivitäten auf neue Produkte für neue Märkte." Eine "laterale Diversifikation bedeutet den Vorstoß in völlig neue Produkt- und Marktgebiete, wobei die Unternehmung aus dem Rahmen ihrer traditionellen Branche ausbricht und in weitabliegenden Aktivitätsfeldern tätig wird."

[Meff98]

**Dublin Core**

Allgemeine Bezeichnung für ein System zur Bereitstellung von Internet-basierten Metadaten für elektronische und physische Ressourcen, das insbesondere für das Suchen und Indizieren dieser Ressourcen dienen soll. Konkrete Ausprägung ist das Dublin Core Metadata Element Set (DCMES).

nach [DCMI01a]

**Dublin Core Metadata Element Set**

Erster konkreter Standard der DCMI zur Beschreibung zentraler Metainformationen wie "Urheber", "Datum" oder "Beschreibung" einer Ressource mittels einheitlicher Attribute. Als RFC 2413 der IETF veröffentlicht.

nach [DCMI01a]

**Dublin Core Metadata Initiative**

"The Dublin Core Metadata Initiative (DCMI) is an organization dedicated to fostering the widespread adoption of interoperable metadata standards and promoting the development of specialized metadata vocabularies for describing resources to enable more intelligent resource discovery systems."

[DCMI01a]

**Electronic Data Interchange**

Standardisiertes Verfahren zum elektronischen Austausch von strukturierten Geschäftsdaten wie Bestellungen, Rechnungen usw.

**Extensible Markup Language**

W3C-Standard einer erweiterbaren Beschreibungssprache auf Basis von SGML.

**File Transfer Protocol**

Protokoll zur Übertragung von Dateien in TCP/IP-basierten Computernetzwerken.

nach [Tane97]

**Hit**

Einzelner Request an einen Webserver. Der Abruf einer einzigen HTML-Seite (Pageview) kann mehrere Hits erzeugen, wenn die Seite Frames oder Grafiken enthält.

**Hypertext Markup Language**

W3C-Standard einer Beschreibungssprache für WWW-Seiten (Hypertext) auf Basis von SGML.

nach [Tane97]

**Internet Engineering Task Force**

Organisation zur Entwicklung und Etablierung von Internetprotokollen und -standards.

**Internet Protocol**

Grundlegendes Protokoll zur Übertragung von Datenpaketen im Internet und anderen IP-Netzen. Aufgrund der engen Verzahnung mit den darauf aufbauenden Transportprotokollen UDP und insbesondere TCP wird häufig vom TCP/IP-Protokoll(-Stack) gesprochen.

nach [Tane97]

**Life-Time-Value**

Höhe der durch einen Kunden erreichten kumulierten Wertschöpfung für ein Unternehmen über die Dauer der gesamten Kundenbeziehung.

**Pageview**

"Sichtkontakt mit einer Webseite (...), unabhängig von der Anzahl der enthaltenen Elemente beziehungsweise aufgerufenen Dateien."

[Stol99]

**Platform for Privacy Preferences**

W3C-Standard zum Schutz der Privatsphäre von Internetnutzern.

**Post Office Protocol, Version 3**

Protokoll zur Übertragung von e-Mails in TCP/IP-basierten Computernetzwerken, insbesondere zum Abruf von e-Mails vom Mailserver durch den Client.

nach [OHE99]

**Remote Method Invocation**

Entfernter Prozedur-Aufruf bei Java.

**Request for Comment**

Die Veröffentlichung eines Vorschlags für einen Internet-relevanten Standard zur Diskussion unter involvierten Entwicklern. RFCs werden von der IETF verwaltet und sind nach der Reihenfolge ihres Erscheinens durchnummeriert.

**Session**

Auch: Server Session. Eine Menge von Pageviews, die zu einer Serie von HTTP-Requests an einen einzelnen Webserver gehören und von einem einzelnen Nutzer verursacht wurden.

nach [MCS00]

**Short Message Service**

Mobilfunkdienst zur Übertragung textueller Kurznachrichten zwischen Mobiltelefonen.

**Simple Mail Transfer Protocol**

Protokoll zur Übertragung von e-Mails in TCP/IP-basierten Computernetzwerken, insbesondere zum Versand von e-Mails und zur Weiterleitung zwischen Mail-Servern.

nach [OHE99]

**Standard Generalized Markup Language**

Standard einer Meta-Beschreibungssprache, welche die Ableitung spezieller Beschreibungssprachen (z. B. HTML, XML) erlaubt.

**Structured Query Language**

Standardisierte Sprache für Datenbanken zur Definition, Manipulation und Anfrage von Daten.

**Transmission Control Protocol**

Zuverlässiges verbindungsorientiertes Transportprotokoll auf Basis des Internet Protocol.

nach [Tane97]

**Unified Modeling Language**

"The Unified Modeling Language (UML) is a language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems."

[OMG00]

**Uniform Resource Identifier**

Zeichenkette zur Identifikation von Ressourcen im Internet.

**Uniform Resource Locator**

Untermenge der URIs, die Informationen über die Zugriffart auf eine Internetressource enthalten, also beispielsweise die Protokollart 'http://' oder 'ftp://'.

**Up-Selling**

Strategie beim Verkauf, bei der Kunden der Erwerb höherklassiger Produkte als ursprünglich geplant nahegelegt wird. (vgl. Cross-Selling).

**User Datagram Protocol**

Unzuverlässiges verbindungsloses Transportprotokoll auf Basis des Internet Protocol.

nach [Tane97]

**Voting**

Online durchgeführte Abstimmung von Nutzern über eine Fragestellung oder Bewertung eines Sachverhalts. Das Ergebnis wird dem Nutzer i. d. R. nach Stimmabgabe angezeigt oder kann zur Empfehlung von Inhalten dienen.

**Wireless Application Protocol**

Offener Protokollstandard für die Übertragung von Daten von und zu mobilen Endgeräten wie Mobiltelefonen und Handheld-Computern.

**World Wide Web**

Internet-basierter Anwendungsdienst zur Anzeige multimedial angereicherter, verknüpfter Texte (Hypertext) unter Verwendung von HTTP und HTML. Umgangssprachlich oft synonym mit Internet verwendet.

nach [Tane97]

**World Wide Web Consortium**

Unabhängige Organisation zur Koordination und Weiterentwicklung von Internetstandards. (Vgl. <http://www.w3.org>)

## LITERATURVERZEICHNIS

- [ATG00]** ATG: *Dynamo 4 Programmer's Guide*  
Version von 2000-03-02  
Art Technology Group, Inc.
- [Balz00]** Balzert, Helmut: *Lehrbuch der Software-Technik*  
Bd. 1 – Softwareentwicklung, 2. Aufl., Spektrum, Heidelberg 2000
- [Baum00]** Baumgarten, Matthias et al.: *User-Driven Navigation Pattern Discovery from Internet Data*  
In: Masand, B.; Spiliopoulou, M. (Ed.): *Web Usage Analysis and User Profiling*, S. 74-91  
Springer, Berlin/Heidelberg 2000
- [Best00]** Bestavros, Azer: *Next Generation Profiling & Personalization Strategies*  
<http://www.personalization.com/summit/londonreview/AzerBestavros-Allaire%20Corp.ppt>  
(2001-01-08)
- [BoHo00]** Bohrer, Kathy; Holland, Bobby: *Customer Profile Exchange (CPExchange) Specification Version 1.0*  
[http://www.cpexchange.org/standard/cpexchangev1\\_0F.pdf](http://www.cpexchange.org/standard/cpexchangev1_0F.pdf)  
(2001-01-10)
- [Broa99]** BroadVision Educational Services: *AppDev 1 – Schulungunterlagen*  
Version 4.1, Oktober 1999  
BroadVision, Inc.
- [CDA00]** Cingil, Ibrahim; Dogac, Asuman; Azgin, Ayca: *A Broader Approach to Personalization*  
In: *Communications of the ACM*, Vol. 43, No. 8, S. 136-141 – August 2000
- [ChJe00]** Chen, Yao-Tsung; Jeng, Bingchiang: *A Customer Segmentation Mining System on the Web Platform*  
In: *American Conference on Information Systems 2000*, S. 36-40
- [Coco99]** Coco, Joe et al.: *U sing P3P for E-Commerce*  
W3C Note, 1999-11-29  
<http://www.w3.org/TR/1999/NOTE-P3P-for-ecommerce-19991129>  
(2001-03-04)
- [CLM01]** Cranor, Lorrie; Langheinrich, Marc; Marchiori, Massimo: *A P3P Preference Exchange Language 1.0 (APPEL1.0)*  
W3C Working Draft, 2001-02-26  
<http://www.w3.org/TR/2001/WD-P3P-preferences-20010226>  
(2001-03-04)
- [Cran00]** Cranor, Lorrie et al.: *The Platform for Privacy Preferences 1.0 (P3P1.0) Specification*  
W3C Candidate Recommendation, 2000-12-15  
<http://www.w3.org/TR/2000/CR-P3P-20001215/>  
(2001-01-28)
- [DCMI01a]** Dublin Core Metadata Initiative: *DCMI Frequently Asked Questions (FAQ)*  
<http://dublincore.org/resources/faq/index.shtml>  
(2001-03-08)
- [DCMI01b]** Dublin Core Metadata Initiative: *Dublin Core Metadata Element Set, Version 1.1: Reference Description*  
<http://dublincore.org/documents/1999/07/02/dces/>  
(2001-03-08)

- [DCMI01c]** Dublin Core Metadata Initiative: *Dublin Core Qualifiers*  
<http://dublincore.org/documents/2000/07/11/dcmes-qualifiers/>  
(2001-03-08)
- [Duch99]** Duchrow, Martin: *Virtuelle Communities – Die Konsumgesellschaften der Zukunft?*  
In: Hermanns, Arnold; Sauters, Michael (Hrsg.): *Management-Handbuch Electronic Commerce*, S. 435-450  
Vahlen, München 1999
- [ECIN01]** Electronic Commerce Info Net: *B2B-Umsätze: Wo laufen sie denn?*  
<http://www.ecin.de/marktbarometer/b2b-umsatz/index.html>  
(2001-01-18)
- [ECML01a]** ECML.org: *ECML (1.0) Specifications*  
<http://www.ecml.org/specifications.html>  
(2001-03-04)
- [Fiel99]** Fielding, R. et al.: *Hypertext Transfer Protocol -- HTTP/1.1*  
RFC 2616  
<ftp://ftp.isi.edu/in-notes/rfc2616.txt>  
(2001-02-20)
- [FiKo00]** Fields, Duane K.; Kolb, Mark A.: *Web Development with Java Server Pages*  
Manning Publications, Greenwich 2000
- [FoSc00]** Fowler, Martin; Scott, Kendall: *UML konzentriert*  
2. Aufl., Addison-Wesley, München 2000
- [Fowl99]** Fowler, Martin: *Analysemuster – Wiederverwendbare Objektmodelle*  
Addison-Wesley, München 1999
- [Gart00]** Gartmann, Rüdiger: *Anforderungen an eine Plattform für multimediale Electronic Commerce-Dienste*  
Diplomarbeit DII, Lehrstuhl Wirtschaftsinformatik der Produktionsunternehmen, Universität-Gesamthochschule Essen
- [GHJV96]** Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John: *Entwurfsmuster. Elemente wiederverwendbarer objektorientierter Software*  
Addison-Wesley, München 1996
- [GrTh00]** Gruhn, Volker; Thiel, Andreas: *Komponentenmodelle. DCOM, JavaBeans, Enterprise JavaBeans, CORBA*  
Addison-Wesley, München 2000
- [HaAr97]** Hagel III, John; Armstrong, Arthur G.: *Net Gain. Profit im Netz*  
Th. Gabler Verlag, Wiesbaden 1997
- [HaCo96]** Hallam-Baker, Phillip M.; Connolly, Dan: *Session Identification URI*  
W3C Working Draft, 1996-02-21  
<http://www.w3.org/pub/WWW/TR/WD-session-id-960221.html>  
(2001-02-26)
- [Hand01]** Handschumacher, Sven: *Eine Referenzarchitektur für komponentenbasierte e-Business-Anwendungen*  
Diplomarbeit, Lehrstuhl X, Fachbereich Informatik, Universität Dortmund
- [Hens97]** Hensley, Pat et al.: *Proposal for an Open Profiling Standard*  
W3C Note, 1997-06-02  
<http://www.w3.org/TR/NOTE-OPS-FrameWork.html>  
(2001-01-28)

- [HeSa99]** Hermanns, Arnold; Sauters, Michael: *Electronic Commerce - Grundlagen, Potentiale, Marktteilnehmer und Transaktionen*  
In: Hermanns, Arnold; Sauters, Michael (Hrsg.): *Management-Handbuch Electronic Commerce*, S. 13-29  
Vahlen, München 1999
- [Kass00]** Kassem, Nicholas et al.: *Designing Enterprise Applications with the Java 2 Platform, Enterprise Edition*  
Version 1.0.1, Final Release, 2000-10-03  
Sun Microsystems, Palo Alto 2000
- [KoBI99]** Kotler, Philip; Bliemel, Friedhelm: *Marketing-Management. Analyse, Planung, Umsetzung und Steuerung*  
9. Aufl., Schäffer-Poeschel Verlag, Stuttgart 1999
- [KRR00]** Kim, D. J.; Ramesh, M.; Rao, H. R.: *Web Ads Selection for One-to-One Advertising Using Neuro-Fuzzy Systems*  
In: American Conference on Information Systems 2000, S. 26-32
- [KRT99]** Kalakota, Ravi; Robinson, Marcia; Tapscott, Don: *E-Business: Roadmap for Success*  
Addison-Wesley 1999
- [LaSw99]** Lassila, Ora; Swick, Ralph R. (Ed.): *Resource Description Framework (RDF) Model and Syntax Specification*  
W3C Recommendation, 1999-02-22  
<http://www.w3.org/TR/1999/REC-rdf-syntax-19990222>  
(2001-03-08)
- [LHHL97]** Lincke, David-Michael; Haertsch, Patrick; Hoffmann, Christoph P.; Lindemann, Markus A.: *Integrierte Electronic Commerce Systeme - Auswahlkriterien und Evaluation aktueller Produktangebote*  
Arbeitsbericht, Institut für Wirtschaftsinformatik, Universität St. Gallen, November 1997  
[http://www.businessmedia.org/netacademy/publications.nsf/all\\_pk/247](http://www.businessmedia.org/netacademy/publications.nsf/all_pk/247)  
(2001-07-03)
- [LiHa94]** Link, Jörg; Hildebrand, Volker G.: *Verbreitung und Einsatz des Database Marketing und CAS*  
Vahlen, München 1994
- [Lin00]** Link, Jörg: *Database Marketing*  
In: Bliemel, Friedhelm; Fassot, Georg; Theobald, Axel (Hrsg.): *Electronic Commerce. Herausforderungen – Anwendungen – Perspektiven*, S. 106-122  
3. Aufl., Th. Gabler Verlag, Wiesbaden 2000
- [MAB00]** Mulvenna, Maurice D.; Anand, Sarabjot S.; Büchner, Alex G.: *Personalization on the Net using Web Mining*  
In: Communications of the ACM, Vol. 43, No. 8, S. 123-125 – August 2000
- [MaSp00]** Masand, Brij; Spiliopoulou, Myra: *Web Usage Analysis and User Profiling*  
Springer, Berlin/Heidelberg 2000
- [MaHa99]** Matena, Vlada; Hapner, Mark: *Enterprise JavaBeans™ Specification, v1.1*  
Release 1999-12-17  
[ftp://ftp.java.sun.com/pub/ejb/11final-129822/ejb1\\_1-spec.pdf](ftp://ftp.java.sun.com/pub/ejb/11final-129822/ejb1_1-spec.pdf)  
(2000-04-10)
- [MCS00]** Mobasher, Bamshad; Cooley, Robert; Srivastava, Jaideep: *Automatic Personalization Based on Web Usage Mining*  
In: Communications of the ACM, Vol. 43, No. 8, S. 142-151 – August 2000

- [Meff98]** Meffert, Heribert: *Marketing. Grundlagen marktorientierter Unternehmensführung*  
8. Aufl., Th. Gabler Verlag, Wiesbaden 1998
- [Moba00]** Mobasher, Bamshad et al.: *Discovery of Aggregate Usage Profiles for Web Personalization*  
<http://robotics.stanford.edu/users/ronnyk/WEBKDD2000/papers/bamshad.pdf>  
(2001-07-10)
- [OHE99]** Orfali, Robert; Harkey, Dan; Edwards, Jeri: *Client/Server Survival Guide*  
3rd Ed., John Wiley & Sons, New York 1999
- [OMG00]** Object Management Group: *OMG Unified Modeling Language Specification – Version 1.3*  
<http://www.omg.org/cgi-bin/doc?formal/00-03-01.pdf>  
(2000-06-01)
- [PeEt00]** Perrowitz, Mike; Etzioni, Oren: *Adaptive Web Sites*  
In: Communications of the ACM, Vol. 43, No. 8, S. 142-151 – August 2000
- [PeRo96]** Peppers, Don; Rogers, Martha: *Strategien für ein individuelles Kundenmarketing. Die 1:1 Zukunft.*  
Droemer, München 1996
- [PRD99]** Peppers, D.; Rogers, M.; Dorf, B.: *The One to One Field Book*  
Capstone Publishing Limited 1999
- [Röde00]** Röder, Holger: *Electronic Commerce und One-to-One-Marketing*  
In: Bliemel, Friedhelm; Fassot, Georg; Theobald, Axel (Hrsg.): *Electronic Commerce. Herausforderungen – Anwendungen – Perspektiven*, S. 145-158  
3. Aufl., Th. Gabler Verlag, Wiesbaden 2000
- [Roma99]** Roman, Ed: *Mastering Enterprise JavaBeans and the Java 2 Platform, Enterprise Edition*  
John Wiley & Sons, New York 1999
- [Scho00]** Schonberg, Edith et al.: *Measuring Success*  
In: Communications of the ACM, Vol. 43, No. 8, S. 53-57 – August 2000
- [SpEc99]** Spainhour, Stephen; Eckstein, Robert: *Webmaster in a Nutshell*  
2nd Ed., O'Reilly & Associates, Sebastopol 1999
- [Spil00]** Spiliopoulou, Myra: *Web Usage Mining for Web Site Evaluation*  
In: Communications of the ACM, Vol. 43, No. 8, S. 127-134 – August 2000
- [Stol99]** Stolpmann, Markus: *Online-Marketingmix: Kunden finden, Kunden binden im E-Business*  
Galileo Press GmbH, Bonn 1999
- [Stol00]** Stolpmann, Markus: *Kundenbindung im E-Business: Loyale Kunden – Nachhaltiger Erfolg*  
Galileo Press GmbH, Bonn 2000
- [Tane97]** Tanenbaum, Andrew S.: *Computernetzwerke*  
3. Aufl., Prentice Hall Verlag, München 1997
- [TSB00]** Tomczak, Torsten; Schögel, Marcus; Birkhofer, Ben: *Online-Distribution als innovativer Absatzkanal*  
In: Bliemel, Friedhelm; Fassot, Georg; Theobald, Axel (Hrsg.): *Electronic Commerce. Herausforderungen – Anwendungen – Perspektiven*, S. 219-238  
3. Aufl., Th. Gabler Verlag, Wiesbaden 2000
- [Vlis99]** Vlissides, John: *Entwurfsmuster anwenden*  
Addison-Wesley, München 1999
- [Volo00]** Volokh, Eugene: *Personalization and Privacy*  
In: Communications of the ACM, Vol. 43, No. 8, S. 84-88 – August 2000
- [Walt01]** Walther, Ralf: *Web Mining*  
In: Informatik Spektrum, Band 24, Heft 1, S. 16-18 – Februar 2001