



Projekt mobiCOMP

Jörg Lonthoff

Technische Universität Darmstadt
Fachgebiet Wirtschaftsinformatik I
Entwicklung von Anwendungssystemen
Hochschulstraße 1
64289 Darmstadt
Lonthoff@winf.tu-darmstadt.de
<http://www.mobiCOMP.de>

Zusammenfassung: Die Zukunft der ingenieurmäßigen Softwareentwicklung ist die Komponentenorientierung. In diesem Beitrag wird das Forschungsprojekt mobiCOMP vorgestellt. mobiCOMP ist ein „mobiler Marktplatz“ zum Handel von (Software-)Komponenten. Hierbei wird „mobil“ in zwei Richtungen verstanden. Zum einen werden Komponenten auch für mobile Endgeräte angeboten und zum anderen wird die Komponentenbereitstellung durch Unterstützung mobiler (drahtloser) Kommunikationskanäle ermöglicht.

Ein mobiler Marktplatz kann der zentrale Ort zur Beschaffung von „mobilen“ Komponenten werden. Denn dort stehen die Angebote dem Nachfrager konzentriert zur Verfügung. Vorstellbare Anwendungsszenarien reichen vom Marktplatz für Midlets (ressourcenschonende Softwarespielereien, wie Handy-Spiele oder Klingeltöne) bis hin zum automatisierten Handel und Konfigurieren von Fachkomponenten (z. B. Kreditprüfung) für den Aufbau komplexer Anwendungssysteme im mobilen Umfeld.

1 Einleitung

Elektronische Marktplätze haben sich im B2B-Bereich etabliert. Durch Rechnerunterstützung der Kaufphasen werden Transaktionskosten nachweislich gesenkt [Week01]. Auch der Komfort für die Marktteilnehmer (=Anbieter und Nachfrager) wird deutlich dadurch gesteigert, dass sich Wege verkürzen, Märkte konzentrieren, die Auswahl erhöht und durch Standardisierungsbemühungen der Austausch an Informationen vereinheitlicht wird.

Unsere Welt wird schrittweise mobiler. Es begann mit der räumlichen Mobilität durch die Nutzung von Eisenbahn, Auto und Flugzeug. Im Internetzeitalter wurden die angebotenen Dienste zunächst auf stationär mit dem Netzwerk verbundenen Rechner angeboten. Im nächsten Schritt der Mobilitätsentwicklung werden (Sprach-)Handlungen durch mobile Endgeräte, wie Handy und PDA elektronisch unterstützt. Es vollzieht sich also eine Erweiterung des E-Commerce zu einem M-Commerce [Roth02, S. 8]. In diesen Bereichen siedeln sich elektronische bzw. mobile Marktplätze an.

„Mobile Marktplatzsysteme“ sind von Vorteil, wenn der Marktzugang ortsunabhängig bzw. geräteunabhängig erfolgen soll. Bestehende elektronische Marktplatzsysteme haben den Nachteil, dass sie entweder gar nicht oder nur schlecht mobil nutzbar sind. Denn der



Versuch, elektronische Marktplätze nur durch die Transformation der Inhalte auf mobile Endgeräte „mobilfähig“ zu machen, ist nicht ausreichend, da nicht jedes Gerät für die Darstellung bestimmter Inhalte gleich gut geeignet ist. So genügt eine reine Protokollübersetzung zwischen z. B. html und wml nicht, um Inhalte im Internet von einer für einen gewöhnlichen Rechner geeigneten Darstellung in eine für das Handy geeignete zu transformieren, da das Handy-Display deutlich kleiner ist als ein Computer-Bildschirm. Manchmal ist eine Darstellung nicht zwingend notwendig, da Komponenten auch ohne Benutzerinteraktion direkt vom Gerät automatisiert abgerufen werden können. (z. B. Update). In einem solchen Fall muss die Darstellung umgangen werden. Auch sind Netzwerkverbindungen im mobilen Fall wesentlich anfälliger auf Fehler als im stationären Fall, ja es muss sogar mit einem Totalausfall für eine bestimmte Zeit bzw. für bestimmte Orte (Netzabdeckung, bzw. Tunnel) gerechnet werden.

Mit dem laufenden Forschungsprojekt mobiCOMP wird ein „mobiler Marktplatz“ entwickelt, der von jedem „Ort“ bzw. mit jedem möglichen Zugangsgeschäft erreichbar ist. Darüber hinaus sollen Komponenten gebrauchstauglich angeboten werden. Ein „mobiler Marktplatz“ kann der zentrale Ort zur Beschaffung von „mobilen“ Komponenten werden. Denn dort stehen die Angebote dem Nachfrager konzentriert zur Verfügung. Vorstellbare Anwendungsszenarien reichen vom Marktplatz für Midlets (ressourcenschonende Softwarespielereien, wie Handy-Spiele oder Email-Clients) bis hin zum Handel von Fachkomponenten (z. B. Kreditprüfung) für den Aufbau komplexer Anwendungssysteme.

In diesem Beitrag werden zunächst die Begriffe „Mobilität“, „elektronischer Marktplatz“, „mobiler Marktplatz“ und „Komponente“ geklärt. Anschließend werden Anforderungen an mobile Marktplätze hergeleitet. Danach wird der aktuelle Entwicklungsstand des Projekts mobiCOMP vorgestellt, Anwendungsszenarien beschrieben und Vorschläge für dessen Weiterentwicklung gegeben. Daran schließt sich eine Diskussion des Beitrags, sowie Fazit und Ausblick an.

2 Mobilität und Märkte

2.1 Mobilität

Intuitiv bedeutet Mobilität – auch geistiger Art – sich bewegen können. Worauf bezieht sich diese Mobilität? *Geistige Mobilität* kann als die Fähigkeit eines Anwenders verstanden werden, über den Einsatz verfügbarer Schemata zu reflektieren, neue Schemata zu erwerben und bestehende Schemata, wenn dies geboten erscheint, zu modifizieren. Im Umfeld des Mobile Computing unterscheidet Roth drei Arten von Mobilität: Endgerätemobilität, Benutzermobilität und Dienstmobilität [Roth02, S. 7].

Endgerätemobilität ist bei einem PC nicht gegeben. Obwohl ein solcher Rechner durchaus „mobil“ ist – man kann ihn tragen –, wird er doch während des Herumtragens nicht funktionieren. Dazu müsste er mit Akkumulatoren, Display und kabellosem Netzwerkzugang ausgestattet sein, aber dann würde er wohl eher Laptop genannt. Somit impliziert die Endgerätemobilität ein Gebrauchen des Gerätes im nicht stationären Fall. Zentral sticht das Merkmal der ortsunabhängigen Benutzung hervor. Eine einfache und mühelose Herstellung und Aufrechterhaltung der Gerätemobilität für eine angemessene Zeit sind damit Anforderungen an ein „mobiles Endgerät“.

Bei der *Benutzermobilität* liegt der Fokus auf einem sich bewegendem Benutzer, der sich zwar durchaus stationär aufhalten kann, aber nicht an einen besonders ausgezeichneten Ort bzw. an ein bestimmtes Gerät gebunden ist. Hier ist also der Benutzer mobil.

Auch Dienste können als „mobil“ aufgefasst werden, wenn sie sich von Gerät zu Gerät übernehmen lassen, oder wenn sie von überall aus erreicht werden können. Eine solche Form der Mobilität bezeichnet man als *Dienstmobilität*.

2.2 Elektronischer Marktplatz

Ein Marktplatz im klassischen Sinne ist ein bestimmter Ort, an dem Anbieter und Nachfrager zusammenkommen. Beim elektronischen (virtuellen) Marktplatz spielen Ort und Zeit keine Rolle, denn es handelt sich hierbei um einen „virtuellen Raum“, der nur mit Hilfe elektronischer Geräte bzw. dem Einsatz von Informationstechnologie von allen Teilnehmern „betreten“ werden kann [Koll02, S. 39]. In diesem Sinne müssen die Marktteilnehmer zum Marktplatz-Zugangsgerät (z. B. PC, der an das Internet angeschlossen ist) hingehen.

Oft wird vom Markt mit vollständiger Konkurrenz gesprochen. Damit sind folgende Eigenschaften verbunden: atomistische Angebots- und Nachfragestruktur, Fehlen von Präferenzen bzw. Diskriminierungen von Anbietern und Nachfragern, vollständige Markt- bzw. Preistransparenz, unbegrenzte Mobilität sämtlicher Produktionsfaktoren und Güter [Koll02, S. 41].

Elektronische Marktplätze erfüllen sicherlich nicht das Ideal der vollständigen Konkurrenz, können sich diesem im Vergleich zu herkömmlichen Märkten jedoch weitaus besser nähern: die Markt- und Preistransparenz ist größer, jeder kann sich schneller informieren; durch die Möglichkeit des anonymen oder pseudonymen Auftretens im Netz können Diskriminierungen vermieden werden. Gerade mit fortschreitender Durchdringung des Alltags mit Technik wird der Zugang zum elektronischen Markt immer leichter.

Ein elektronisches Marktplatzsystem ist ein vollständiges System, das Nachfragern und Anbietern ermöglicht, alle notwendigen Handelstransaktionen abzuschließen. Dabei lassen sich grob drei Phasen einer Handelstransaktion identifizieren: Informations-, Verhandlungs- und Abwicklungsphase [Merz02, S. 30ff].

Des Weiteren können Marktplätze bzgl. des verwendeten Marktmechanismus in Katalog-, Börsen- und Auktionsmärkte eingeteilt werden. Eine gute Einführung in elektronische Märkte findet sich in [Koll01].

2.3 Mobiler Marktplatz

Ein mobiler Marktplatz ist ein elektronischer Marktplatz, der Dienstmobilität und Benutzermobilität unterstützt. Im Sinne der Dienstmobilität muss der Marktplatz eindeutig adressierbar sein, damit er unabhängig vom Ort des Benutzers erreichbar ist. Zur Unterstützung der Benutzermobilität sollte ein mobiler Marktplatz den Zugang durch mobile Endgeräte unterstützen. Dafür müssen drahtlose Kommunikationsmedien, z. B. in Form einer WAP-Schnittstelle, bereitgestellt werden. Das mobile Marktplatzsystem selbst ist



nicht notwendigerweise physisch mobil im Sinne der Endgerätemobilität, sondern wird üblicherweise auf einem oder mehreren verteilten Servern betrieben.

Der Gedanke dabei ist, dass ein Marktteilnehmer nicht mehr zum Markt hin gehen muss, sondern der Markt stattdessen zum Nachfrager bzw. auf dessen Zugangsgerät kommt.

2.4 Komponenten

Nach Szyperski zeichnet sich eine Komponente durch drei charakteristische Eigenschaften aus (vgl. [Szyp98]):

- eine Komponente ist eine unabhängig einsetzbare Einheit,
- eine Komponente wird von Dritten mit anderen Komponenten zu einem Kompositum verbunden und
- der Zustand einer Komponente ist nicht persistent.

Um als unabhängig einsetzbare Einheit gelten zu können, muss eine Komponente von ihrer Umgebung, insbesondere anderen Komponenten, klar abgegrenzt werden. Sie kapselt ihre Funktionalität und stellt diese nur über vertraglich festgelegte, definierte Schnittstellen nach außen bereit. Dabei sind die Softwarekontrakte meist als Vor- und Nachbedingungen einer über eine Schnittstelle angebotenen Funktion oder als deren Invariante zu verstehen. Wird die Komponente in einen Kontext gestellt, der ihren Vor- und Nachbedingungen genügt, so erfüllt sie ihren Teil des Vertrages. Dritte – dies sind in der Regel nicht die Hersteller der Komponenten und haben keinen Zugriff auf ihre Implementierungsdetails – fügen verschiedene Komponenten zu einem Kompositum, einer Anwendung, zusammen (vgl. [Szyp98, S. 34]).

Eine andere Definition des Begriffs wird von Ackermann et al. vorgenommen (vgl. [Acke⁺02, S. 1]). Eine Komponente besteht aus verschiedenartigen (Software-) Artefakten. Sie ist wiederverwendbar, abgeschlossen und vermarktbar, stellt Dienste über wohldefinierte Schnittstellen zur Verfügung, verbirgt ihre Realisierung und kann in Kombination mit anderen Komponenten eingesetzt werden, die zur Zeit der Entwicklung nicht unbedingt vorhersehbar sind.

Über die Definition von Szyperski hinausgehend, werden hier alle Artefakte, die einer Komponente zuzuschreiben sind, eingeschlossen, also neben der Komponente im Binärformat auch ihre Dokumentation, in ihr verankerte Grafiken, Voreinstellungen oder Testfälle. Zudem wird mit der Eigenschaft der Vermarktbarkeit verlangt, dass die Komponente als eigenständiges wirtschaftliches Gut beispielsweise auf offenen aber auch unternehmensinternen Marktplätzen gehandelt werden kann. In den übrigen Merkmalen gleichen sich die beiden angeführten Definitionen.

Ackermann et al. führen neben dem allgemeinen Begriff der (Software-)Komponente noch den Begriff „Fachkomponente“ ein (vgl. [Acke⁺02, S. 1]). Eine Fachkomponente ist eine Komponente, die eine bestimmte Menge von Diensten einer betrieblichen Anwendungsdomäne anbietet.

Darüber hinaus können noch weitere Komponentenarten unterschieden werden [Acke+02]. Komponenten lassen sich in einfache, aber dennoch abgeschlossene, und zusammengesetzte Komponenten einteilen. Neben den bereits erwähnten Fachkomponenten zählen die



Systemkomponenten, die generische Funktionen realisieren, zu den einfachen Komponenten. Die zusammengesetzten Komponenten lassen sich weiter in anpassbare und fertige unterteilen. Fertig zusammengesetzte Komponenten sind dann zum einen Baugruppen, die aus System- und oder Fachkomponenten zusammengesetzt sind und auch Konfigurationen genannt werden, und Anwendungen, die als große, unabhängig einsetzbare Komponenten aufzufassen sind, jedoch auch noch mit anderen Komponenten verbunden werden können. Ein Komponenten-System-Framework erfüllt generische, ein Komponenten-Anwendungs-Framework anwendungsspezifische Funktionen. Beide lassen sich als zusammengesetzte Komponenten aber anpassen, indem sie beispielsweise um weitere einfache oder auch zusammengesetzte Komponenten zu einer fertigen Anwendung erweitert werden.

Das Paradigma der komponentenorientierten Anwendungsentwicklung beschreibt Ortner wie folgt (vgl. [Ortn03b, S. 109]):

Die Komponenten werden aus einem Katalog entnommen und dann nach einem Plan (z.B. einer Syntax, auf der Basis eines Frameworks oder mit Hilfe von Stücklisten und Konstruktionsplänen) zusammengesetzt.

Damit wird ein Paradigmenwechsel von der Entwicklung monolithischer Standardsoftware-Pakete für einzelne Unternehmensfunktionen, die in Grenzen an die Bedingungen ihres Einsatzbereiches angepasst werden können (customizing), aber häufig auch organisationale Veränderungen in dem die Software nutzenden Unternehmen notwendig machen, hin zur Entwicklung individuell gestaltbarer, flexibler Anwendungen vollzogen, die sich aus Komponenten zusammensetzen, die auf Märkten gehandelt werden und nur dann bei ihrer Einführung in ein Unternehmen zu organisationalen Änderungen führen, wenn diese zum Erreichen des Effizienzoptimums des ergonomischen Systems beitragen.

3 Anforderungen an mobile Marktplätze

Als theoretische Grundlage dieses Forschungsprojekts wurden in Studien- und Seminararbeiten spezifische Anforderungen an mobile Anwendungssysteme, speziell an „mobile Marktplätze“ herausgearbeitet, die sich in die folgenden Bereiche einteilen lassen:

- Kommunikation
 - Art: verbindungsorientiert/verbindungslos, drahtgebunden/drahtlos
 - Verfügbare Bandbreite
 - Fehleranfällige Kommunikation -> Fehlertolerante Mechanismen
- Endgeräteheterogenität
 - Darstellungsanpassung
 - Speicherplatz, Betriebssystem, Rechenleistung -> Bereitstellung von Geräteprofilen
- Lokalisierbarkeit

Auf eine detailliertere Darstellung wird aus Ressourcengründen bewusst verzichtet. Einzelne Aspekte können in der Standardliteratur, z. B. bei [Roth02], [Hans+01] und [Lehn03], nachgelesen werden.

Aus diesen Anforderungen entstand dann die mobiCOMP Marktplatzarchitektur, die in diesem Beitrag vorgestellt wird. Auf dieser Grundlage können Marktplatzsysteme schon beim Entwurf „mobil gemacht“ werden.

4 mobiCOMP

Das Projekt mobiCOMP entstand als logische Fortentwicklung der vergangenen zwei Wirtschaftsinformatikpraktika am Fachgebiet Wirtschaftsinformatik I – Entwicklung von Anwendungssystemen der TU Darmstadt. 2001 wurde ein elektronischer Marktplatz für Softwarekomponenten mit dem Namen CompoNEX, im darauf folgenden Jahr mit mobileISM eine Servicemanagementplattform für mobile Endgeräte auf PocketPC-Basis entwickelt. mobiCOMP bringt die mobile Welt und den Komponentenhandel in einem integrierten System zusammen.

mobiCOMP ist ein mobiler Marktplatz zum Handel von (Software-)Komponenten im mobilen Umfeld. Seine grundlegenden Funktionen sind das Anbieten und Suchen, sowie das Kaufen von Komponenten. Der Anbieter stellt seine Komponente und deren Beschreibung, die gemäß des GI-Spezifikationsrahmens [Acke+02] in einem Katalog abgelegt wird, in das Marktplatzsystem. Der Nachfrager kann Komponenten entweder über dessen Namen oder hierarchisch über Kategorien suchen. Eine Detailansicht der Komponente liefert die durch den Anbieter bereitgestellte Beschreibung. Ein kaufwilliger Nachfrager kann dann die Komponente kaufen und herunterladen.

Im aktuellen Entwicklungsstand dient mobiCOMP zunächst als Intermediär, über den allein lauffähige, unabhängige Softwarekomponenten angeboten werden.

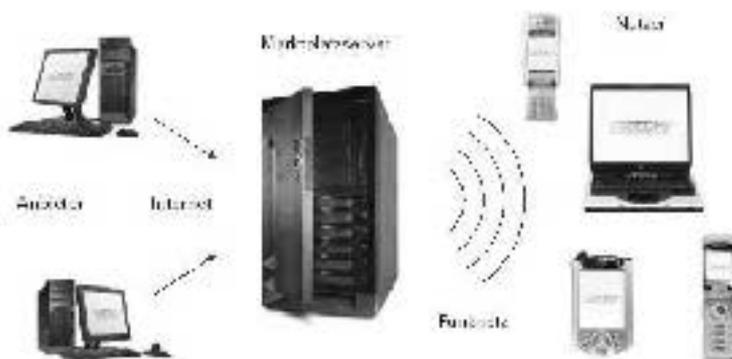


Abbildung 1: mobiCOMP – Hardware [Quelle: eigene Darstellung]

Als mobile Marktzugangsgeräte werden PDAs, die einen Internetbrowser bereitstellen, und alle Handys mit einem WAP-Browser unterstützt. Bei den angebotenen Komponenten

handelt es sich um kleine Java-Applikationen. Somit laufen diese Komponenten auf allen Endgeräten, auf denen eine Java VM zur Verfügung steht. Als Kommunikationskanal dienen Internet (LAN/WLAN), WAP über GSM, GPRS und UMTS, sowie Bluetooth.

In diesem Ausbaustadium wurde mobiCOMP auf der CeBIT 2004 präsentiert.

4.1 mobiCOMP-Architektur

Als Architekturgrundlage diente die vom Fachgebiet Wirtschaftsinformatik I entwickelte E-NOgS-Architektur [OrOv03]. Somit besteht das mobiCOMP Marktplatzsystem selbst aus einzelnen funktionalen Komponenten, die als Web Services auf Microsoft .NET-Basis implementiert sind. Diese Kapselung bietet ein hohes Maß an Erweiterbarkeit und Wiederverwendbarkeit. Dadurch wird eine nahtlose Integration in die Anwendungsumgebung z. B. eines Unternehmens ermöglicht. Das folgende Schaubild verdeutlicht den komponentenorientierten Systemaufbau von mobiCOMP.

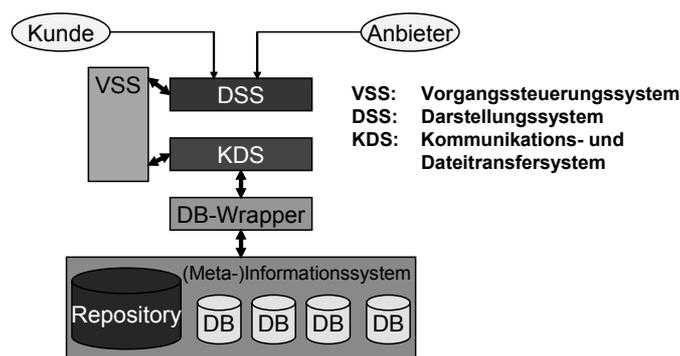


Abbildung 2: mobiCOMP – Software [Quelle: eigene Darstellung]

4.1.1 Das Darstellungssystem (DSS)

Das Darstellungssystem ist die generische grafische Benutzerschnittstelle des Marktplatzsystems. Dieses System sorgt dafür, dass die Darstellung der zu übermittelnden Informationen auf das jeweilige Endgerät, das mit dem Marktplatz verbunden ist, angepasst wird. So wird ein Handy-Benutzer mit Informationen im WML-Format versorgt, die dann sein Endgerät optimal mit dem WAP-Browser darstellen kann. Für einen PDA, der ein viel kleineres Display besitzt als ein PC, wird ebenfalls eine reduzierte Darstellung im HTML-Format generiert. Um eine solche Anpassung der Darstellung vernünftig in den Griff zu bekommen, ist es notwendig, Darstellung und Inhalt zu trennen. Hierbei ist XML das Mittel der Wahl. Über eine Transformation mittels XML Stylesheet Language Transformation (XSLT) können dann die Inhalte gefiltert und mittels Stylesheets auflösungsabhängig

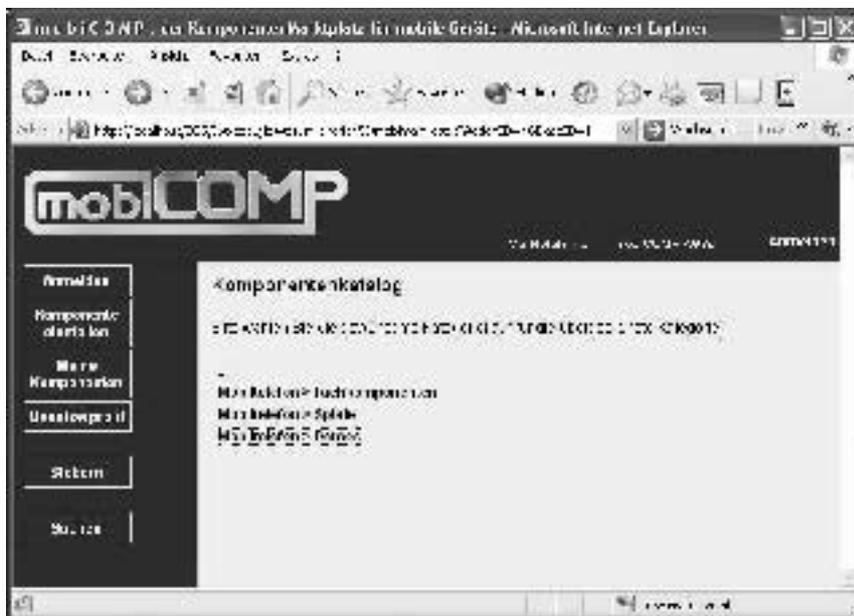


Abbildung 3: Darstellung von mobiCOMP in einem Internet Browser [Quelle: Screenshot]

transformiert werden. Die anzuzeigenden Bildelemente werden dann in Formularen angeordnet und gespeichert.

Darüber hinaus sorgt das DSS für eine Überprüfung der getätigten Eingaben. Überprüft werden Feldtyp, Wertebereich und das Ausfüllen von Pflichtfeldern. Sollte mindestens eine der Eingaben ungültig sein, so wird das Formular mit einem entsprechenden Hinweis erneut angezeigt. Sobald alle Eingaben korrekt vorliegen, werden diese an das VSS weitergeleitet. Alle Vorgaben zur Überprüfung werden den Metadaten der Felder entnommen. Die Felder werden in Formularen angeordnet. Die Formularmetadaten enthalten alle auf einem Formular anzuzeigenden Felder mit deren Metadaten. Diese Metadaten sind komplett in der Datenbank hinterlegt und werden jedem an das DSS gelieferte Objekt angefügt.

Aktionen des Benutzers werden an das VSS gesendet. Dieses verarbeitet die Aktion und sendet eine Formularidentifikationsnummer zurück. Daraufhin wird vom DSS das anzuzeigende Formular gerätespezifisch generiert.

4.1.2 Kommunikations- und Dateitransfersystem (KDS)

Das KDS ist das zentrale Kommunikationssystem. Es dient als einheitliche Kommunikationsschnittstelle zur Weiterleitung der Anfragen des Vorgangssteuerungssystems. Nach außen sorgt das Dateitransfersystem für die Sicherstellung einer einwandfreien und vollständigen Übertragung der Komponente. Dafür muss das KDS feststellen können, ob ein Transfer vollständig abgewickelt wurde. Im Fehlerfall muss das VSS informiert werden.

4.1.3 Vorgangssteuerungssystem (VSS)

Das VSS übernimmt die Koordination und Steuerung aller möglichen Aktionen des Marktplatzsystems in allen Transaktionsphasen.

In der Informationsphase müssen die notwendigen Informationen zur Verfügung gestellt werden. So muss der Nachfrager Komponenten suchen und deren Details anschauen können. In der Verhandlungsphase besteht die große Herausforderung in einer Rechnerunterstützung der Vertragsverhandlung. In der Abwicklungsphase muss das VSS dafür sorgen, dass alle relevanten Arbeitsschritte bei der Abwicklung eines geschlossenen Vertrages eingehalten werden. Nach der Feststellung, dass ein Kaufvertrag abgeschlossen wurde, muss die Zahlung des Kaufpreises erfolgen. Die Zahlungsinformation kann – je nach Geschäftsmodell und Art des Zahlungsverkehrs – von Kreditinstituten, dem Verkäufer der Komponente oder aus internen Quellen kommen. Nach erfolgter Zahlung wird dann im Regelfall die Komponente zum Endverbraucher transferiert. Dieser Transfer muss dann auch sichergestellt sein. Ein Verbindungsabbruch darf nicht zu einem Verlust der erworbenen Komponente beim Nachfrager führen, sondern es müssen evtl. Wiederholungsversuche des Komponenten-Downloads gestattet sein. Dabei ist der Komponententransfer Aufgabe des KDS. Das VSS übernimmt also die Ablauflogik des Systems, welche im Moment noch statisch im Programmcode verankert ist.

4.1.4 Datenbankwrapper (DBW)

Der Datenbankwrapper kapselt den Zugriff auf die Datenbanken. Der Grund für die zentrale Koordination aller Abfragen an das (Meta-)Informationssystem liegt darin, dass in diesem Fall eine Änderung der Datenbankstruktur nicht zu Modifikationen des Programmcodes führt. Der DBW stellt drei Funktionen zur Verfügung: Insert, Select und Delete. Dabei werden die Tabellenstrukturen und die Objekte aus XML-Dateien ausgelesen, um die Abfragen entsprechend generisch aufzubauen.

Angemerkt sei an dieser Stelle, dass der DBW lediglich den technischen Zugang zu den Datenbanken auf eine logische, abstraktere Stufe stellt und den anderen Komponenten somit den Zugriff auf alle Datenspeicherungssysteme erleichtert. Er hat dafür Sorge zu tragen, dass die Anfragen in der Sprache der Datenbank(en) syntaktisch korrekt formuliert sind und die Antworten der Datenbanksysteme in die Sprache des anfragenden Systems zur Verfügung gestellt werden.

4.1.5 (Meta-)Informationssystem

Die Komponente, die in der funktionalen Übersicht mit (Meta-)Informationssystem bezeichnet ist, besteht aus einem Repository (DBMIS), in dem die Metadaten abgelegt werden und den Datenbanken DBKat, DBUser, DBMEg, DBTrans, DBError und DBNorm.

DBKat – Komponentenkatalog-Datenbank

DBKat umfasst die zur Umsetzung des Komponentenspezifikationsrahmens der GI [Acke+02, S. 4] notwendigen Relationen. Es wurde eine möglichst weitgehend mit der

Theorie übereinstimmende Realisierung angestrebt, sodass das vorliegende Marktplatzsystem zugleich als „Durchführbarkeitsstudie“ des vorgeschlagenen Spezifikationsrahmens gelten kann.

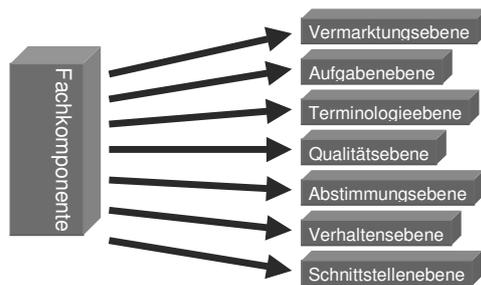


Abbildung 4: Beschreibungsebenen von Fachkomponenten [Acke+02, S. 4]

- **Vermarktungsebene**

Die Vermarktungsebene beinhaltet im wesentlichen vier Kategorien von Daten: grundlegende Informationen über abzulegende Komponenten, wie Komponentennamen und -version, Informationen über Bezugsmodalitäten (bspw. Komponentenpreis, Kontaktdaten und Vertragsbedingungen), klassifizierende Daten, welche die hierarchische Kategorisierung nach Wirtschaftszweig und Anwendungsdomäne ermöglichen, sowie Anforderungen an die zugrunde liegende Systemarchitektur (zu untergliedern in Hardware, Software und Basissysteme).

- **Aufgabenebene**

Die Aufgabenebene erfasst normsprachlich die Funktionalität der Komponente, wobei von den einzelnen, an der Schnittstelle bereitgestellten Prozeduren, abstrahiert wird. Aufgaben können hierbei hierarchisch (sowohl abstraktiv als auch kompositiv) beliebig zerlegt und von entsprechenden Prozeduren der Schnittstelle referenziert werden. Hierbei wird von DBNorm Gebrauch gemacht, welche die Normsprachenfunktionalität zur Verfügung stellt.

- **Terminologieebene**

Die Terminologieebene des GI-Spezifikationsrahmens ist Teil der Normsprache und demgemäß in DBNorm enthalten.

- **Qualitätsebene**

Die Qualitätsebene erlaubt die hierarchische Speicherung von Qualitätsklassen mit zugeordneten Qualitätseigenschaften, welche wahlweise für eine gesamte Komponente oder für einzelne Komponentenfunktionen gelten kann. In beiden Fällen werden Qualitätsausprägungen gespeichert, welche sich zwecks Vergleichbarkeit explizit auf im System vorgegebene Referenzumgebungen beziehen.

- **Abstimmungsebene**

Die Abstimmungsebene hat zum Ziel, die Reihenfolgebeziehungen zwischen den Funktionen der Komponenten in formaler Notation (TemporalOCL) zu spezifizieren.

- **Verhaltensebene**

Die Verhaltensebene spezifiziert in formaler Notation (Object Constraint Language (OCL)) die Vor- und Nachbedingungen sowie Invarianten der Komponenten.

- **Schnittstellenebene**

Die Schnittstellenebene umfasst neben der direkt von Programmierumgebungen verwendbaren Interface Definition Language (IDL) der Komponenten die Zuordnung von IDL-Funktionen zu Aufgaben der Aufgabenebene sowie von IDL-Datenobjekten zu Termini der Terminologieebene, wodurch semantische Korrektheit gewährleistet werden kann. Eine IDL-Funktion kann hierbei mehreren Aufgaben zugeordnet werden, ein Datenobjekt jedoch nur mit einem Terminus übereinstimmen.

DBUser – Benutzer-Datenbank

DBUser speichert sämtliche dem System bekannten natürlichen und juristischen Entitäten strukturiert ab. Hierbei wird unterschieden zwischen natürlichen Personen (Datenstruktur „Person“), welche sowohl als Systembenutzer (verbunden mit einem Login-Account) als auch als Ansprechpartner für Unternehmen (Datenstruktur „ContactPerson“ als Resultat der Konnexion mit „Company“) fungieren können und juristischen Personen, welche zur Identifikation des Herstellers der Komponenten dienen.

DBMEg – Endgeräte-Datenbank

DBMEg dient der strukturierten, hierarchischen Gliederung der vom Marktplatzsystem unterstützten mobilen Endgeräte. Neben der Einordnung des jeweiligen Gerätetyps unter eine der hierarchisch gegliederten Geräteklassen (Handy, Smartphone, PDA) werden Informationen, wie Bildschirmauflösung und Farbtiefe, abgelegt, die zur Darstellung der Marktplatz-Bedienungsumgebung auf dem Endgerät sowie der Unterstützungsfunktionalität bezüglich Downloads von anderen Marktplatzkomponenten verwendet werden.

DBTrans – Transaktionen-Datenbank

DBTrans dient als operative Logbuch-Datenbank; jegliche systemrelevanten Ereignisse werden hier chronologisch – zu Debugging-Zwecken, als Datenfundament für Statistiken und für Sicherheitsabfragen (bspw. versuchtes Mehrfach-Einloggen) etc. – protokolliert.

Systemrelevante Ereignisse lassen sich untergliedern in benutzer-, komponenten- und systemorientierte Ereignisse. Benutzerorientierte Ereignisse sind Login und Logout von Benutzern, Registrierung und Abmeldung vom Marktplatz sowie Änderungen von Benutzerdaten. Komponentenorientierte Ereignisse sind das Auswählen der Detailansicht einer Komponente, Kauf und Download von Komponenten, Einfügen, Ändern und Löschen von Komponenten. Eine strukturierte Gliederung der im Gesamtsystem möglichen Fehlerfälle ist den systemorientierten Ereignissen zuzuordnen (siehe hierzu auch DBError).

DBError – Fehler-Datenbank

Ziel von DBError ist die zentralisierte Administration sämtlicher potenziell möglicher Fehlerereignisse inkl. Gliederung in ihrerseits baumstrukturartig festgehaltenen Fehlerklassen.

DBNorm – Normsprachenschema

Die Normsprachenebene stellt eine ontologiebasierte Funktionalität für zwei Sprachebenen bereit: Auf der objektsprachlichen Ebene (Sprachebene „1“) werden Termini und Satzbaupläne zur Beschreibung von im Marktplatz gehandelten Komponenten abgelegt, auf der 2. Sprachebene hingegen werden metasprachliche Konstrukte (Attribute, Relationen) Termini zugeordnet, sodass u.a. transparente Synonymitätsauflösung ermöglicht wird.

Zur Erreichung o.g. Ziele erfolgt eine explizite, schrittweise und zirkelfreie Definition von Fachbegriffen (Termini), welche nicht mit den weiter unten aufgeführten Schlüsselbegriffen (keywords) zu verwechseln sind: Während erstere in der Aufgabenebene verwendet werden und i.d.R. Datenstrukturen von Komponenten normalsprachlich bezeichnen, dienen Schlüsselbegriffe der semantischen Suche nach Komponenten. Definiert werden Termini nach den gängigen Definitionsregeln – neben obligatorischer Kurzdefinition sind Langdefinitionen, Prädikatorenregeln und Verwendungsbeispiele möglich. Des Weiteren können Termini zueinander in Beziehungen (z.B. „*ist Gattung von*“ oder „*ist Schlüssel für*“) stehen.

Objektsprachliche Termini können entweder global – dies ist insbesondere zur Standardisierung und der daraus resultierenden Interoperabilitätsermöglichung wichtig – oder lokal für einzelne Komponenten gelten. Metasprachliche Termini sind in diesem Sinne stets als global anzusehen.

DBMIS – Metainformationssystem

DBMIS stellt Metainformationen, d.h. Informationen über die im System existierenden Schemata (Konzepte wie Relationen, Attribute, Integritätsbedingungen etc.) strukturiert dar. Hierbei erfolgt eine transparente Auflösung von Synonymen auf Attributebene durch Bezugnahme auf normsprachliche Daten (DBNorm).

4.2 Anwendungsszenarien

Vorstellbare Anwendungsszenarien reichen vom Marktplatz für Midlets bis zum automatisierten Handel und Konfigurieren von Fachkomponenten für den Aufbau komplexer Anwendungssysteme.

Szenario Handy-Spielereien:

In diesem Szenario werden auf mobiCOMP so genannte Midlets gehandelt. Das sind Klingeltöne, Bilder, Betreiberlogos, Spiele und ressourcenschonende Anwendungen. Der Markt dafür ist vorhanden, d.h. es gibt schon etliche Anbieter (z. B. Jamba!) und die Nachfrage ist groß. Der Vorteil von mobiCOMP besteht darin, dass das Produkt nicht mehr über teure 0190er Nummern oder umständliche Codes zu erhalten ist, sondern entweder direkt per WAP-Portal oder über einen Bluetooth-Hotspot, z.B. bei einem Telekommunikationsladen oder an Tankstellen.

Szenario mobile Business:

Auf dem Marktplatz werden einzelne Fachkomponenten für mobile Endgeräte gehandelt. Das können z. B. Komponenten für den mobilen Zugang zu ERP-Systemen sein oder Komponenten für die Abrechnung von mobilen Dienstleistungen.

Szenario mobiles externes Anwendungsmanagement:

In diesem Szenario, welches die höchste Entwicklungsstufe eines abstrakten Komponentenmarktplatzes darstellt, werden benötigte (Fach-)Komponenten automatisiert vom Marktplatzsystem abgerufen, konfiguriert und auf dem mobilen Endgerät installiert. Bei dieser Vision ist das Marktplatzsystem das externe Anwendungsmanagement für das mobile Endgerät.

5 Weiterentwicklung

Ein hochgestecktes Ziel ist eine rechnerunterstützte benutzerfreundliche Bereitstellung von Anwendungen auf Komponentenbasis. Dabei soll benutzerfreundlich so verstanden werden, dass ein „echter Benutzer“, der kein spezielles Vorwissen im IT-Umfeld hat, den Marktplatz bedienen kann und durch das Angebot auf dem Marktplatz unterstützt wird.

Eine Erleichterung für den Benutzer ist das automatische Vorfiltern des Angebots. Dies kann z. B. dadurch geschehen, dass beim Zugriff auf den Marktplatz das Endgerät des Benutzers identifiziert wird und automatisch Komponenten für diese Gerätekategorie vorgeschlagen werden. Bei dieser Erweiterung sollte eine explizite Suche nach ausgeblendeten Angeboten trotzdem noch möglich sein, falls das Gerät falsch identifiziert wurde oder der Benutzer explizit etwas anderes sucht.

Die Unterstützung semantischer Suchen anhand von Stichwörtern erleichtert das gezielte Auffinden von Komponenten insbesondere im Bereich mobiler Endgeräte, welche aufgrund gegebener Geräterestrictionen keine komplexen Suchanfragen gestatten. Stichwörter besitzen hierbei eine „Nähe“ zueinander, sodass von eingegebenen Stichwörtern zu „Nachbarn“ gemäß den im System abgelegten Distanzmaßen übergegangen werden kann (*fuzzy search*). Dies wäre eine sinnvolle Erweiterung der Suche bei expliziten Anfragen durch den Benutzer.

Ein Marktplatz kann verschiedene Rollen einnehmen. Als *Intermediär* ist er z. B. zwischen Produzent und Kunden anzuordnen. Damit nimmt er die Funktion einer Anbieter-Nachfrager-Zusammenführung wahr, ohne dass Produzenten und Kunden direkt zusammentreffen. Im Sinne eines *Integrators* (vgl. [VoZe03, S. 222]) tritt er auf, wenn durch das Marktplatzsystem unabhängige, physisch und logisch getrennte, meist heterogene Systeme miteinander verbunden werden. Wenn die Funktionen der Komponenten als „Dienste“ bereitgestellt werden, wird aus dem Marktplatzsystem ein so genannter *Trader* (vgl. [EiSc03, S. 303]), da die Dienste in einem verteilten System vermittelt werden. Darüber hinaus kann ein Marktplatz durch Bereitstellung von zusätzlichen Diensten als *Service-center* einen Mehrwert bereitstellen. Im Rahmen des laufenden Forschungsprojekts soll mobiCOMP als Trader und Integrator weiter ausgebaut werden.

Das Ziel einer generischen, regelbasierten Marktplatzsteuerung konnte noch nicht erreicht werden. Für eine flexible Ablaufsteuerung ist das Vorgangssteuerungssystem derart zu erweitern, dass es ausschließlich generischer Natur ist und die anzuwendenden Regeln (Prozessschemata) einer Datenbank entnimmt. Durch eine solche flexible Steuerung des Marktplatzsystems können auch unterschiedliche Marktformen und Marktmechanismen unterstützt werden ohne das Gesamtsystem modifizieren zu müssen.



Um die Qualität der angebotenen Komponenten zu gewährleisten, wird ein Prüfstand für Komponenten entwickelt. Bei erfolgreichem Projektabschluss wird dieser in den mobiCOMP-Marktplatz integriert.

Für die automatisierte Komposition von Komponenten wird eine Composer-Komponente benötigt. Diese hat zur Aufgabe, die Komponenten anwendungs- und engerätespezifisch zusammenzustellen und so zu konfigurieren, dass das Kompositum automatisch auf dem Endgerät installiert wird.

Ein Marktplatz, der diese Automatisierung von Konfiguration und Installation beherrscht, kann den kompletten Komponenten-Life-Cycle managen, in dem er auch noch Funktionen zum automatischen Deinstallieren, bzw. Aktualisieren von Komponenten anbietet.

Zur Bereitstellung einer sicheren Übertragung von Komponenten kann eine Kryptografie-Komponente die Verschlüsselung des Komponententransfers übernehmen.

Eine orthogonale Funktionserweiterung von mobiCOMP kann dahingehend erfolgen, dass die Lokalisierungsmöglichkeit von Benutzern mit berücksichtigt wird und entsprechende Location Based Services (LBS) angeboten werden.

6 Diskussion



Grundsätzlich ist die Frage erlaubt, wozu mobile Marktplätze nützlich sind, oder gar ob überhaupt von mobilen Marktplätzen gesprochen werden darf. Unter den Aspekten der Benutzer- und Dienstmobilität hat der Begriff seine Berechtigung.



Im Probetrieb des Prototypen wird die Architektur des Systems evaluiert und muss gegebenenfalls überarbeitet werden.

Die Zukunft der ingenieurmäßigen Softwareentwicklung ist die Komponentenorientierung. Doch heute gibt es noch gar nicht so viele Komponenten, die auf Märkten gehandelt werden. Hier muss auf Seiten der Anbieter noch vieles getan werden.

7 Fazit und Ausblick

Mit mobiCOMP liegt ein Prototyp eines mobilen Marktplatzsystems vor, der unter Verwendung von XML und Web Services auf Basis von Microsoft .NET im Rahmen eines Wirtschaftsinformatikpraktikums implementiert wurde. Im Rahmen eines Dissertationsvorhabens wird mobiCOMP nun konzeptionell weiterentwickelt.

Literatur

- [Acke⁺02] Ackeremann, J.; Brinkop, F.; Conrad, S.; Fettke, P.; Frick, A.; Glistau, E.; Jaekel, H.; Kotlar, O.; Loos, P.; Mrech, H.; Ortner, E.; Raape, U.; Overhage, S.; Sahm, S.; Schmiedendorf, A.; Teschke, Th.; Turowski, K.: Vereinheitlichte Spezifikation von Fachkomponenten. Memorandum des Arbeitskreises 5.10.3 der Gesellschaft für Informatik e.V. „Komponentenorientierte betriebliche Anwendungssysteme“, Februar 2002.



- [EiSc03] Eicker, S.; Schwichtenberg, H.: Universal Component Trading – Trading in heterogenen Komponentenumgebungen. In (Uhr, W.; Esswein, W.; Schoop, E. Hrsg.): Wirtschaftsinformatik 2003/Band I – Medien, Märkte, Mobilität, Physica-Verlag, Heidelberg, 2003; S. 303-323.
- [Hans⁺01] Hansmann, U.; Merk, L.; Nicklous, M.; Stober, T.: Pervasive Computing Handbook, Springer Verlag, Berlin, 2001.
- [Koll01] Kollmann, T.: Virtuelle Marktplätze: Grundlagen – Management – Fallstudie, München, 2001.
- [Lehn03] Lehner, F.: Mobile und drahtlose Informationssysteme: Technologien, Anwendungen, Märkte, Springer Verlag, Berlin, 2003.
- [Merz02] Merz, M.: E-Commerce und E-Business: Marktmodelle, Anwendungen und Technologien, 2. aktualisierte und erweiterte Auflage, dpunkt.verlag, Heidelberg, 2002.
- [Ortn03] Ortner, E.: Sprachingenieurwesen – Wie man mit Wörtern die Cyber-Welt bewegt. Arbeitsbericht 2003/01 Fachgebiet Wirtschaftsinformatik I – Entwicklung von Anwendungssystemen, Technische Universität Darmstadt, 2003.
- [OrOv03] Ortner, E.; Overhage, S.: E-NOgS: Ein komponentenorientiertes Middleware-Framework für E-Commerce-Anwendungen. In: (Wörner, J. Hrsg.): thema Forschung 1/2003, Verlag für Marketing und Kommunikation, Monsheim, 2003.
- [Roth02] Roth, J.: Mobile Computing: Grundlagen, Technik, Konzepte, 1. Auflage, dpunkt.verlag, Heidelberg, 2002.
- [Szyp98] Szyperski, C.: Component Software: Beyond Object-Oriented Programming, Addison-Wesley, Harlow, 1998.
- [VoZe03] Voigtmann, P.; Zeller, T.: Beiträge zur Integrationsproblematik im Kontext von Electronic Business und Elektronischen Marktplätzen. In (Uhr, W.; Esswein, W.; Schoop, E. Hrsg.): Wirtschaftsinformatik 2003/Band I – Medien, Märkte, Mobilität, Physica-Verlag, Heidelberg, 2003; S. 215-237.
- [Week01] Week, S.M.: Management of Engineering and Technology, 2001. PICMET '01. Portland International Conference on, Volume: 1 , 29 July-2 Aug. 2001, Pages:149 vol.1.