



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Seminarausarbeitung

Alewtina Schumann

Intelligente Software-Agenten in E-Commerce

Alewtina Schumann
Intelligente Software-Agenten in E-Commerce

Seminarausarbeitung im Rahmen der Veranstaltung Anwendungen II
im Studiengang Informatik (Master)
am Fachbereich Elektrotechnik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

bei : Prof. Dr. Kai von Luck

Abgegeben am 21. Februar 2007

Alewtina Schumann

Thema der Seminausarbeitung

In intelligente Software-Agenten in E-Commerce

Stichworte

Agenten, Software-Agenten, E-Commerce

Kurzzusammenfassung

In dieser Ausarbeitung wird der Einsatz und das Nutzen der intelligenten Software-Agenten im elektronischen Handel betrachtet. Es wird kurz auf die Technologie der Agenten eingegangen und einige Asätze aus dem E-Commerce angeschaut, wo Agenten angesetzt werden.

Inhaltsverzeichnis

Abbildungsverzeichnis	5
1 Einführung	6
1.1 Bezug zum Projekt	7
1.2 Zielsetzung und Gliederung	7
2 Grundlagen der Software-Agenten	8
2.1 Definition	8
2.2 Komponenten und Architektur	9
2.3 Kommunikation	11
3 Agentenanwendungen in Electronic Commerce	12
3.1 Elektronische Auktionen	12
3.1.1 AuctionBot	13
3.2 Elektronischer Marktplatz	14
3.2.1 Trading Agent Competition	14
4 Zusammenfassung	16
Literaturverzeichnis	17

Abbildungsverzeichnis

2.1	Interrap-Architektur nach Müller (1996)	10
3.1	Architektur von AuctionBot	13
3.2	Architektur von Mertakor	15

1 Einführung

Eigentlich war Electronic Commerce einer der üblichen Einsatzgebiete für die Agententechnologie. Agent-mediated e-Commerce (AMEC) befasst sich mit auf Agenten basierten Lösungen, die unterschiedlichen Stufen des Handelprozesses unterstützen. Die Umwelt des E-Commerce ist inzwischen komplexer und dynamischer geworden. Mit der Spezialisierung der Endgeräte und ihrem Anschluss an das Internet entsteht die Möglichkeit zur spontanen Vernetzung, die von Technologien wie JAVA, JINI oder Bluetooth unterstützt wird. Eine Vielzahl mobiler Endgeräte übernimmt die Aufgaben der zentralen Universalmaschine Personal Computer, wobei eine drahtlose Vernetzung überdies eine spontane Kooperation erlaubt. Die Vernetzung betrifft nicht nur die Hardware, sondern wird auch Hardware-Software-Kombinationen miteinander in Verbindung bringen. So tendiert auch die Geschäftswelt in mehreren zusammenhängenden Online-Marktplätzen zu handeln, indem sie neue drahtlose Kommunikationstechnologien und sie unterstützende Geräte nutzt. Es entsteht der Begriff der ortsunabhängigen mobilen E-Commerce (m-Commerce). Hier spielt die Mobilität der Agenten, die es ihnen erlaubt frei durch die Netzwelt zu spazieren, eine wichtige Rolle für dessen Einsatz in neuen E-Commerce-Anwendungen. Diese Eigenschaft erlaubt es den mobilen Agenten, rasch auf die Änderungen auf den Marktplätzen zu reagieren und schneller Entscheidungen zu treffen als ferne Agenten oder der Mensch. Weitere mobile Eigenschaften sprechen für die Nutzung von Agenten in verteilten Systemen [M. Jalali-Sohi \(2001\)](#):

- Entlastung des Netzwerkes: Mobile Agenten können Netzwerkverkehr reduzieren, indem sie sich im Netz bewegen und sich z.B. auf einen Datenserver umlagern können.
- Verbindungsunabhängigkeit: Agenten können ihre Aufgaben ausführen auch wenn der Benutzer nicht mehr mit dem Internet verbunden ist.
- Hohe Stabilität: Mobilität kann benutzt werden, um die Fehlertoleranz zu erhöhen und Stabilität von verteilten Systemen zu unterstützen. Wenn ein Server ausgeschaltet wird oder Probleme meldet, kann ein Agent darauf reagieren, indem er auf einen anderen Server migriert und da seine Aufgaben weiter bearbeitet.
- Server-Flexibilität: Da mobile Agenten ihr eigenen Code herumtragen können und nicht auf die festgelegte Funktionalität vom Server beschränkt sind, bekommen verteilte Systeme mehr Flexibilität und deren Funktionalität kann erweitert werden, wenn die Agenten ihren Code auf dem Server ausführen.

1.1 Bezug zum Projekt

In dem Flughafenprojekt soll ein Indoor-Navigationssystem entwickelt werden. Die Flughafengäste sollen auf dem Flughafengelände schnell und bequem zu ihren Points-of-Interest geführt werden. Das Führen heißt aber nicht nur, dass eine Route von Punkt A nach Punkt B berechnet und angezeigt wird. Dem Fluggast werden auf Grund seines Profils und seiner aktuellen Position auf dem Flughafengelände Geschäfte vorgeschlagen, die für ihn interessant sein könnten. Die Route kann so berechnet werden, dass andere Points-of-Interest mit einbezogen werden (das entscheidet das System mittels Benutzerprofil), auch wenn der Fluggast es nicht explizit dem System befohlen hat. Außerdem kann virtuelle Werbung, die ein Fluggast auf seinem mobilen Gerät erhält, ihn auf eine oder andere Ware aufmerksam machen.

Ein anderer interessanter Aspekt ist es, dass sich viele Menschen gleichzeitig in einem Ort befinden. Hier können verschiedene ortsabhängige (auch virtuelle) Dienste angeboten werden. Es kann eine Bluetooth-Säule sein, wo man ein MP3-Lied oder ein neues Spiel auf sein mobiles Gerät herunterladen kann, aber auch ein Flirt-System, das die Profile der Benutzer vergleicht und diese entsprechend informiert, wenn eine passende Person in der Nähe ist. Eine virtuelle MP3-Tauschbörse, die auf dem gleichen Prinzip funktioniert wie das Flirt-System, kann eine interessante Anwendung im Kontext der ortsabhängigen Dienste sein. Ein Benutzer sucht nach einem Lied und ein anderer hat es bereits und bietet es an. Wenn die beiden in der Nähe sind, werden sie vom System benachrichtigt und es kann ein Tausch z.B. mittels Infrarot oder Bluetooth stattfinden. In diesem Fall spielt auch die Positionserkennung einer Person eine wichtige Rolle. Die Rolle des Anbieters und des Suchenden können die Software-Agenten übernehmen, die nach den "Geschäftspartnern" suchen und miteinander verhandeln.

1.2 Zielsetzung und Gliederung

Ziel dieser Arbeit ist die Untersuchung existierender Projekte, die sich mit den Software-Agenten in Electronic Commerce beschäftigen. Es soll ein Überblick in die Grundlagen der Multi-Agenten-Systeme unter diesem Kontext geschaffen werden.

Im Kapitel 2 werden die Grundlagen der Software-Agenten beschrieben. Dann im Kapitel 3 werden einige ausgewählte Agentenanwendungen vorgestellt und untersucht, dabei werden die Trading Agent Competition und deren Gewinner betrachtet. Anschließend liefert Kapitel 4 eine Zusammenfassung und den Ausblick.

2 Grundlagen der Software-Agenten

2.1 Definition

Sucht man nach einer Definition von Agenten, so findet viele verschiedene Definitionen, die, obwohl Begriff „Autonomie“ vorkommt, jedes mal ein anderes Schwerpunkt hervorheben.

Marvin Minsky beschreibt einen Agenten als „any part or process of the mind that by itself is simple enough to understand-even though the interactions among groups of such agents may produce phenomena that are much harder to understand“ [Minsky \(1986\)](#).

Eine andere Definition liefert Michael Wooldridge. Nach ihm ist ein Agent „a computer system that is capable of independent action on behalf of its user or owner (figuring out what needs to be done to satisfy design objectives, rather than constantly being told)“ [Wooldridge \(2002\)](#). Wooldridge unterscheidet Agenten nach ihren jeweiligen Charakteristika. Dabei spricht er von dem schwachen Agenten-Begriff(englisch weak notion of agency) und dem starken Agenten-Begriff(englisch strong notion of agency). Diese Charakteristika geben eine Abgrenzung der Agenten zu herkömmlichen Software-Objekten. Damit Computer-Programme überhaupt als Software-Agent bezeichnet werden können, müssen sie mindestens die folgenden Eigenschaften erfüllen [Wooldridge \(2002\)](#):

- *Autonomie*: Agenten handeln selbständig und ohne direktes Eingreifen des Anwenders. Dazu können Agenten auf ihren inneren Zustand zurückgreifen und diesen selbständig modifizieren.
- *Soziale Fähigkeiten*: Agenten müssen beispielsweise mit dem Anwender kommunizieren, um neue Anweisungen zu erhalten. Oder sie befragen Datenbanken nach Stichworten. Nicht zuletzt können sie andere Agenten dazu bringen, mit ihnen zu kooperieren und so ein komplexes Problem gemeinsam lösen.
- *Reaktives Verhalten*: Agenten beobachten ständig ihre Umwelt und reagieren zeitnah auf Veränderungen. Beispielsweise muss ein Roboter-Agent sofort einem plötzlich auftauchenden Hindernis ausweichen.
- *Proaktives Verhalten*: Agenten ergreifen die Initiative, um ihre Ziele zu erreichen. Dazu können sie ihre Umwelt verändern, wenn es für ihre Aufgabe hilfreich ist.

- *Zielorientiertheit*: Agenten sind fähig, komplexe Problemstellungen zu lösen. Die Entscheidung, wie das Problem zu lösen ist, bleibt dabei in der Hand des Agenten. Damit dies rational geschieht, müssen sie ein Ziel oder Zielsystem haben.
- *Kontinuität*: Agenten sind mehr als bloße Scripte, die bei Bedarf gestartet werden und sich nach getaner Arbeit wieder beenden. Vielmehr bleiben Agenten für längere Zeit aktiv.

Software-Agenten können weitere Eigenschaften aufweisen wie Mobilität, wenn sie ihre Umgebung verlassen, ein Netzwerk durchwandern und auf einem anderen Rechner weiterzulaufen können, oder Lernfähigkeit, indem sie sich an ihre Umgebung, Arbeitsmethoden und Vorzüge des Anwenders anpassen. Sie können einen Charakter besitzen und mit anderen Agenten interagieren.

2.2 Komponenten und Architektur

Die Technik der Agenten besteht generell aus drei Bausteinen. Die *Sensoren* versorgen den Agenten mit Informationen über seine Umwelt. Als Sensoren dienen Schnittstellen zu Datenbanken, Katalogen, Auktionatoren oder die direkte Kommunikation mit anderen Agenten. In einigen Anwendungen sind durchaus physikalische Sensoren denkbar. Mit Hilfe der *Effektoren* (auch *Aktuatoren* genannt) kann der Agent in seiner Umwelt tätig werden. Durch sie interagiert der Agent mit seiner Umwelt und verändert diese. In einem elektronischen Markt sind Effektoren Nachrichten wie Kauf- und Verkaufsangebote, wodurch Preisinformationen in Umlauf gelangen, die wiederum von anderen Agenten durch deren Sensoren aufgenommen und verarbeitet werden können. Im Zentrum des Agenten befindet sich sein *internes Modell*, das für die gesamte Informationsverarbeitung, die Ziehung von Schlussfolgerungen, Entscheidungsfindung und Lernvorgänge verantwortlich ist. Es bestimmt die Autonomie und das Sozialverhalten der Agenten und beinhaltet eine formalisierte Beschreibung der Umwelt, mit dem der Agent seine aktuelle Situation bewerten, zukünftige Ereignisse antizipieren und seine eigenen Aktionen entsprechend planen kann.

Nach [Maes \(1991\)](#) bezeichnet die Agenten-Architektur eine bestimmte methodische Vorgehensweise, um Software-Agenten zu implementieren. Die Architektur beschreibt dazu die einzelnen Komponenten des Agenten und wie diese Einheiten zusammenarbeiten. Für Intelligente Software-Agenten haben sich drei verschiedene Architekturen herauskristallisiert: deliberative, reaktive, und hybride Agenten-Architekturen. Deliberative Agenten wissen über Ihre Umwelt Bescheid und können ihre Aktionen planen, während reaktive Agenten nur auf ihre Sensor-Eingaben reagieren und sind daher für hoch-dynamischen Umgebungen besser geeignet. Die Vorteile beider Ansätze vereinigen die hybriden Agenten.

Die Abbildung 2.1 zeigt die Interrap-Architektur nach Müller (1996), einen Vertreter der hybriden Agenten. Die Architektur besteht aus drei Schichten: einer verhaltensbasierten Schicht, einer lokalen Planungsschicht und einer kooperativen Planungsschicht. Außerdem gibt es drei Komponenten, die in allen drei Schichten präsent sind: Weltschnittstelle, Wissensbasis und Kontrolleinheit.

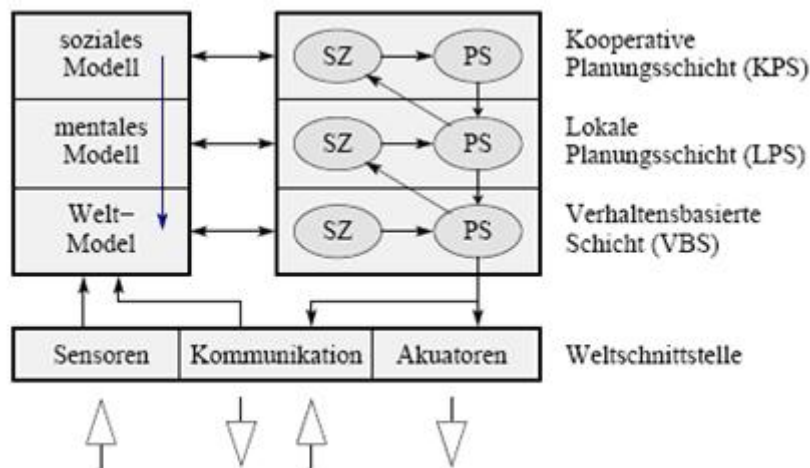


Abbildung 2.1: Interrap-Architektur nach Müller (1996)

Die Weltschnittstelle ermöglicht es dem Agenten mit seiner Umwelt zu interagieren. Dazu enthält diese Komponente die von den reaktiven Agenten bekannten Sensoren und Effektoren. Die Wissensbasis enthält alles Wissen des Agenten und besteht aus den drei Ebenen soziales Modell, mentales Modell und Welt-Modell. Das Welt-Modell enthält Überzeugungen des Agenten über seine Umwelt, beispielsweise dass eine Datenverbindung zu einem anderen Rechnerknoten etabliert werden kann. Das mentale Modell repräsentiert die Überzeugungen des Agenten über sich selbst, beispielsweise dass er fähig ist, bestimmte Problemstellungen zu lösen. Das soziale Modell enthält Überzeugungen des Agenten über andere Agenten, beispielsweise eine Vermutung, wie ein anderer Agent auf ein bestimmtes Angebot reagiert. Die Kontrolleinheit schließlich ist ebenfalls in drei Ebenen eingeteilt, die jeweils eng mit denen der Wissensbasis korrespondieren. Die verhaltensbasierte Schicht (VBS) regelt Routine- und Notfall-Situationen und kann durch ihre reaktive Struktur schnell auf Umweltänderungen reagieren. Übersteigt das Problem die Möglichkeiten der VBS, so gibt sie die Situation an die darüber liegende Schicht weiter. Sowohl die lokale Planungsschicht (LPS) als auch die kooperativen Planungsschicht (KPS) sind deliberativer Natur und besonders für längerfristige Zielfindungen und Planungen erforderlich.

2.3 Kommunikation

Die Multiagentensysteme entstehen, wenn mehrere Agenten miteinander interagieren. Damit verschiedene Agenten, die auch von unterschiedlichen Entwicklern entworfen wurden, sich verstehen können, benötigen sie ein standardisiertes Kommunikationsprotokoll. Ein solches Protokoll stellt die FIPA-ACL (FIPA Agent Communication Language) dar. Wichtige Bestandteile der FIPA-ACL sind:

- **Ontologien:** Legen den Bezug eines Begriffs eindeutig fest. So kann der Begriff „Kohl“ sowohl in den Ontologien Agrarwirtschaft, wie Politik vorkommen.
- **Sprechakte** Innerhalb der FIPA-ACL sind 24 Grammatiken definiert, die einen bestimmten Kommunikationsverlauf festlegen. So legt beispielsweise der Sprechakt „Information“ eindeutig fest, dass der Absender eine Frage stellt und von dem Empfänger darauf eine bestimmte Antwort erwartet. Sicherlich relativ trivial, dennoch aber unbedingt erforderlich.
- **Wissensrepräsentations-Sprachen** Die Darstellung von Informationen (Wissen), wie also z.B. mitgeteilt wird, dass es gerade in Hamburg regnet, wird über sogenannte Wissensrepräsentationssprachen dargestellt. FIPA unterstützt hier gleich einen ganzen Satz solcher Sprachen (z.B. KIF, FIPA-SL, CCL), die dies in unterschiedlicher Syntax tun.

FIPA-ACL definiert einzelne Statements, die im folgenden Beispiel dargestellt sind. In diesem Beispiel möchte Bob von Scott wissen, wie das Wetter bei ihm ist. Scott antwortet, dass es bei ihm regnet:

```
(inform
:sender (agent-identifier :name Scott)
:receiver (set (agent-identifier :name Bob))
:content ("weather(today,raining) ")
:language Prolog
:ontology weather)
```

Der Sprechakt „inform“ definiert Anfrage/Antwort-Schema, das heißt dass ein Agent einen anderen informiert. Auch wenn die Nachricht in einer Sprache (hier Prolog) definiert ist, die nicht jeder Agent versteht, kann ein Agent mittels „receiver“ prüfen, ob die Nachricht an ihn adressiert ist. Im „content“ befindet sich Wissensrepräsentation, also die tatsächliche Nachricht (hier in FIPA-SL). Damit die Agenten sich verstehen müssen sie nicht nur eine gemeinsame Sprache sprechen, sondern sich in einem gemeinsamen Kontext (ontology) unterhalten. Den Kontext der Nachricht liefert der Teil „ontology“ (hier Wetter).

3 Agentenanwendungen in Electronic Commerce

In diesem Kapitel werden aktuelle Arbeiten präsentiert, die sich mit einer ähnlichen Thematik beschäftigen. Ziel dieser Betrachtung ist einen Überblick zu bekommen, wie die Software-Agenten in diesen Projekten eingesetzt werden, und daraus ein Nutzen für das Flughafenprojekt zu gewinnen.

3.1 Elektronische Auktionen

Traditionelle Live-Auktionen finden normalerweise zu einem festgelegten Termin statt, an dem innerhalb weniger Stunden alle Versteigerungspositionen unter den Hammer kommen. Online-Auktionen laufen dagegen häufig viel länger. Fünf bis zehn Tage Dauer sind nicht ungewöhnlich. Ein Artikel wird mit einem Einstiegsgebot und einem Endtermin versehen auf dem Auktionsserver platziert. Ein Bieter kann danach bis zum Endtermin seine Gebote abgeben. Er hat Zeit, das Auktionsangebot zu prüfen, ein erstes Gebot abzugeben und die weitere Preisentwicklung abzuwarten. Sollte sein Gebot im mehrtägigen Verlauf der Auktion überboten worden sein, kann er es bis zum Endtermin jederzeit erhöhen.

Die Gebotsabgabe erfolgt elektronisch. In der Regel gibt der Bieter (nach einer Registrierung vor seiner ersten Auktionsteilnahme) auf einer Internetseite sein vorher vergebenes Passwort und den Betrag ein, den er bieten will, und drückt auf einen Eingabe-Knopf. Daraufhin erhält er per E-Mail eine Bestätigung seines Gebotes. Sollte er im weiteren Verlauf überboten werden, erhält er wieder eine E-Mail-Nachricht mit der Einladung, auf die Auktionsseite zurückzugehen und sein Gebot zu erhöhen. Den Zuschlag erhält, wer zum Endtermin das höchste Gebot abgegeben hat.

Das größte elektronische Online-Auktionshaus Ebay benutzt die Software-Agenten in unterschiedlichen Bereichen des Systems. Es werden beispielsweise Agenten eingesetzt, um die Nutzer zu registrieren. Sind die eingesetzten Agenten durch die Anzahl der Registrierungsanfragen überlastet, erkennen sie dies und setzen eine Replikation in Gang. Um die Anfragen möglichst effizient zu bearbeiten, tauschen sich die Agenten untereinander aus und verständigen sich somit über Status und Vorgehen.

Software-Agenten können nun dem Anwender auch bei Auktionen behilflich sein. Beispielsweise könnte ein Agent eine bestimmte Auktion bewachen und das Gebot des Anwenders erhöhen, wenn er überboten wird. Allerdings gibt es derzeit nur wenig Unterstützung, wenn an mehreren Auktionen in unterschiedlichen Auktionshäusern mitgeboten wird. Beispielsweise könnte ein Anwender daran interessiert sein, fünf Computer zu erwerben. In diesem Fall hat es der Käufer schwer, alle Auktionen zu überwachen und richtig mitzubieten. Im schlimmsten Fall kann er am Ende mehr als fünf Computer erworben haben, was seinen Interessen widerspricht.

3.1.1 AuctionBot

AuctionBot von der Universität von Michigan unterstützt Konsumenten bei verschiedenen Verhandlungsphasen eines Online-Auktions. Das AuctionBot ist ein flexibler und robuster Auktionsserver, der menschliche und Software-Agenten unterstützt [Wurman u. a. \(1998\)](#). Der Server leitet mehrere Auktionen gleichzeitig. Es sammelt Gebote, bestimmt Ergebnisse auf der Basis einer wohl definierten Menge von Auktionsregeln und benachrichtigt Teilnehmer. Es führt keine Transaktionen aus, erzwingt keinen Tausch und überprüft nicht die Kreditwürdigkeit von Auktionsteilnehmern. Benutzer von AuctionBot können sich für eine Auktionsart entscheiden und Auktionsparameter setzen, wie z.B. die Zeitdauer, innerhalb derer konkurrierende Angebote abgegeben werden können, die Methode, wie konkurrierende Gebote aufgelöst werden, Anzahl der zugelassenen Anbieter etc. AuctionBot gestattet den Benutzern einen eigenen Software-Agenten zu kreieren, der autonom auf dem von AuctionBot bereitgestellten Marktplatz mitbietet.

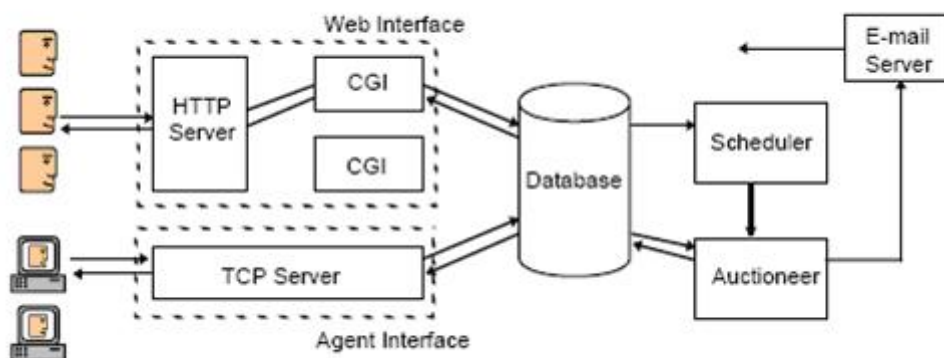


Abbildung 3.1: Architektur von AuctionBot

Das Design des AuctionBot-Konzepts wird in der Abbildung 3.1 dargestellt. Die Agentenschnittstelle setzt sich aus folgenden zwei Teilen zusammen: der Web-Schnittstelle für Menschen und der TCP/IP Schnittstelle für Software-Agenten. Die rechte Seite des Diagramms zeigt die Auktionatorprozesse und den Scheduler. Die Agentenschnittstelle und das Auktionatorprogramm überarbeiten Informationen in einer gemeinsamen Datenbank. Der Scheduler ist ein Prozess, der kontinuierlich die Datenbank nach Auktionen durchsucht, die ein spezielles Ereignis zu bearbeiten, oder ein Gebot zu überprüfen haben. Findet er eine solche Auktion, benachrichtigt er das Auktionärprogramm. Der Auktionär lädt die Auktionsparameter und die Menge der Gebote von der Datenbank. Er bestätigt Gebote und kann bei jedem Durchlauf nur genau ein Gebot löschen oder genau einen Preis benennen. Das Schema basiert darauf, dass die Mehrheit der Auktionen von Menschen benutzt wird, die gewisse Reaktionsgrenzen besitzen. Die Verbindung zwischen der Agentenschnittstelle und dem Scheduler ist asynchron. Wenn der Scheduler offline geht, operiert die Agentenschnittstelle weiter. Die Agentenschnittstelle ist ein TCP/IP Nachrichtenprotokoll, das dem Agenten den Zugang zu allen Merkmalen, präsentiert durch AuctionBot, in der Internetschnittstelle erlaubt. Agenten können Gebote platzieren, Auktionen erstellen, Auktionsinformationen abfragen, oder ihre Accounts einsehen. Das Benachrichtigungsmerkmal, das äquivalent zur Email-Benachrichtigung ist, die in der Internetschnittstelle enthalten ist, ist in dem Agenten API noch nicht enthalten. Das API ist so geschrieben, dass Entwickler es benutzen können, um ihre eigenen Ziele mit AuctionBot zu verfolgen.

3.2 Elektronischer Marktplatz

Ein agentenbasierter elektronischer Marktplatz (AEM) ist ein verteiltes Multiagenten-System, das aus stationären und mobilen Agenten besteht, die den Endbenutzern E-Commerce-Dienste in einem geschäftlichen Kontext anbieten.

3.2.1 Trading Agent Competition

Um die Forschung und Entwicklung der Software-Agenten im elektronischen Markt zu unterstützen findet jährlich die Trading Agent Competition statt. Es ist ein Spiel, wo acht Agenten im Auftrag von ihren Benutzern ein Reisepaket ersteigern sollen. Das Reisepaket besteht aus einem Hin- und Rundflug, einer Hotelreservierung und Unterhaltungsprogramm. Alle Bestandteile des Reisepakets werden in getrennten Auktionen mit unterschiedlichen Regeln versteigert.

Einer der Gewinner dieses Wettbewerbs im Jahre 2005 und 2006 ist Mertacor, ein erfolgreicher Software-Trading-Agent. Die Architektur dieses Agenten verwendet typisches Modul-

Design, das eine synchrone Interkommunikation unterstützt und Änderungen der inneren Bidding-Mechanismen des Agenten ohne großen Aufwand erlaubt [Toulis u. a. \(2006\)](#).

Die drei Überwachungskomponenten Flight, Hotel und Entertainment Auction Einheiten spielen die Rolle des Sensors. Die Informationen, die diese Komponenten vom TAC-Server erhalten (z.B. Preise für die zu ersteigernden Güter), werden durch diese an die Bidder weitergeleitet. Ein weiterer Sensor, der Informationen betreffend Benutzerpräferenzen und über laufende Auktionen erhält, ist die GameInfo-Komponente. Die Bidder-Komponenten repräsentieren drei Mechanismen für das Bieten für jede Auktionsart. Sobald ein Bidder ein neues Angebot berechnet hat, sendet er dieses direkt an den TAC-Server. So repräsentieren Bidder die Effektoren des Mertacors. Mertacor muss sich auf die Arbeit des Linear Programming(LP) Moduls verlassen. Die Komponente gibt optimale Handlungsanweisungen an. Um die besten Entscheidungen für den Erwerb der Güter zu treffen, werden drei verschiedene Methoden verwendet, die auf unterschiedlichen Strategien basieren.

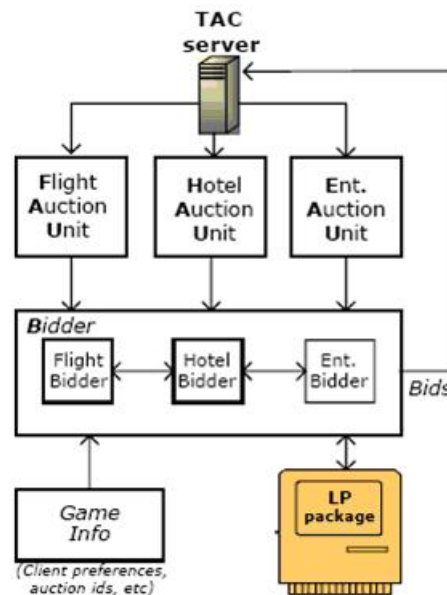


Abbildung 3.2: Architektur von Mertacor

4 Zusammenfassung

Nachdem wir uns einen kurzen Überblick über die Agententechnologie verschaffen und einige Ansätze angeschaut haben, lässt sich feststellen, dass die Software-Agenten sehr gut in das Konzept des elektronischen Handels passen. Das Potenzial ist aber nicht ausgeschöpft.

Im Moment sieht die Situation auf dem E-Commerce -Spielplatz so aus, dass die Kunden die aktive Seite darstellen. Sie suchen gezielt nach Gütern und Diensten, während die Anbieter darauf warten, dass sie von den potenziellen Kunden angesprochen werden. Andererseits wissen die Kunden nicht, wie sie reagieren sollen, wenn ein Angebot, das sie eventuell interessieren könnte, unerwartet auftritt und eine spontane Kooperation entstehen kann.

Agenten können helfen diese Situation zu ändern, indem sie beide Verhandlungsseiten repräsentieren. Man könnte weiter gehen und einen Agenten sowohl für das Kaufen als auch für das Verkaufen einsetzen. Agenten würden den elektronischen Markt flexibler machen und unerwartete Verhandlungen unterstützen. Für die beiden Parteien bedeutet das, dass sie sich um viele Sachen nicht mehr kümmern sollen (z.B. nach neuen Kunden suchen oder Auktionen beobachten) und dadurch ihre Ausgaben senken können.

Literaturverzeichnis

- [ebXML:2005] : *Enabling A Global Electronic Market: ebXML-Official Webpage*. – URL <http://www.ebxml.org>
- [Braun und Rossak 2005] BRAUN, Peter ; ROSSAK, Wilhelm ; DPUNKT.VERLAG (Hrsg.): *Mobile Agents:Basic Concepts, Mobility Models & the Tracy Toolkit*. Morgan Kaufman, 2005
- [Eymann 2001] EYMANN, Torsten: Wenn Mensch und Maschine ein Team bilden: Digital Business Agents und ihre technischen Grundlagen. In: *Diebold Management Report* 1 (2001)
- [Fortino u. a. 2006] FORTINO, Giancarlo ; GARRO, Alfredo ; RUSSO, Wilma: E-Commerce Services based on Mobile Agents, URL citeseer.ist.psu.edu/731406.html, 2006
- [Guttman u.a. 1998] GUTTMAN, Robert H. ; MOUKAS, Alexandros G. ; MAES, Pattie: Agent-mediated electronic commerce: a survey. In: *Knowl. Eng. Rev.* 13 (1998), Nr. 2, S. 147–159
- [Kowalczyk u. a. 2002] KOWALCZYK, Ryszard ; ULIERU, Mihaela ; UNLAND, Rainer: Integrating mobile and intelligent agents in advanced E-commerce: A survey. In: SPRINGER, Berlin (Hrsg.): *3rd International Symposium on Multi-Agent Systems, Large Complex Systems and E-Businesses (MALCEB'2002)* Bd. 2592. Erfurt, Deutschland, 2002, S. 295–313
- [M. Jalali-Sohi 2001] M. JALALI-SOHI, Rainer M.: An Application of Intelligent and Mobile Agents for Retail Visual Mining and E-Commerce on the Internet. In: *COTIM-2001, Conference on Telecommunications and Information Markets*, 2001
- [Maes 1991] MAES, P.: *The agent network architecture*. ACM Press. 1991
- [Mehandjiev und Abdoessalam 2004] MEHANDJIEV, N. ; ABDOESSALAM, A.M: Cooperative Booking of Time-Dependent Travel Services. In: *AAMAS04 Workshop on Agents for Ubiquitous Computing (UbiAgents)*. New York, USA, 2004
- [Minsky 1986] MINSKY, Marvin ; BOOK, A T. (Hrsg.): *The Society of Mind*. Simon&Schuster, 1986
- [Müller 1996] MÜLLER, J.: *The Design of Intelligent Agents: a Layered Approach*. Springer Verlag. 1996

- [Sierra 2004] SIERRA, Carles: Agent-Mediated Electronic Commerce. In: *Autonomous Agents and Multi-Agent Systems* 9 (2004), November, Nr. 3, S. 285–301
- [Toulis u. a. 2006] TOULIS, Panos ; KEHAGIAS, Dionisis ; MITKAS, Pericles A.: Mertacor: A Successful Autonomous Trading Agent. In: *AAMAS06*. Hokkaido, Japan, May 8-12 2006
- [Wooldridge 2002] WOOLDRIDGE, Michael: *An Introduction to MultiAgent Systems*. Chichester, England : John Wiley & Sons, 2002
- [Wurman u. a. 1998] WURMAN, Peter R. ; WELLMAN, Michael P. ; WALSH, William E.: *The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents*. 2d International Conference on Autonomous Agents (Agents-98). May 1998