



O07 im Cyberspace

Softwareagenten und Multiagentensysteme

Fast täglich gibt es in der Informatik neue Ideen und Entwicklungen – und damit auch neue Begriffe und Konzepte. Auch das Gebiet der Softwareagenten und der Multiagentensysteme gehört seit dem Ende der 80er Jahre zu dieser Kategorie von Neueinführungen. Seitdem hat sich diese Disziplin als eine der wichtigsten und stärksten Forschungsdisziplinen innerhalb der Informatik weltweit etabliert. Aber was ist wirklich neu an der Agententechnologie – zum Beispiel im Vergleich zu normalen Softwarepaketen, Objekten oder Expertensystemen. **Softwareagenten: Revolutionäre Helfer oder alter Wein in neuen Schläuchen?**

Mein Name ist Bond. James Bond. Der nonchalante Geheimagent mit dem Faible für schöne Frauen löst bekanntlich jeden Auftrag perfekt. Dass er dabei gelegentlich eigene Wege geht – wen stört es, solange das Resultat stimmt. Der smarte Romanheld im Auftrag Ihrer Majestät hat einiges mit seinen Kollegen aus dem Cyberspace gemein: Auch Softwareagenten sind intelligente, autonome Problemlöser, die auf einem spezifischen Gebiet Aufgaben, oft im Auftrag eines Auftraggebers, bearbeiten.

James Bond ist war im Prinzip ein Befehlsempfänger, löst seine Aufgaben jedoch immer auf seine Art. Agenten sind sogar offiziell *autonom*. Ein Beispiel aus dem prallen Leben: Sie sitzen vor dem Fernseher und schauen sich das Endspiel der Fußball-WM an. Wie immer ist 90 Minuten lang

nichts passiert, so dass Sie dem Bier und den Chips im Übermaß zugesprochen haben. Ausgerechnet jetzt, wo es in der Verlängerung nach Elfmeterschießen aussieht, ist Ihnen beides ausgegangen.

Kein Problem – schließlich hat man seinen persönlichen Agenten, den man direkt losschickt, um die Vorräte aufzustocken. Im täglichen Leben klappt das, wenn der persönliche Assistent entsprechend erzogen oder motiviert ist. Bei einem Softwareagenten funktioniert das nicht. Als Angehöriger der autonomen Szene entscheidet er frei, wie er das von ihm verinnerlichte Ziel erreicht.

Anders ausgedrückt: Man kann einen Agenten nicht zwingen, etwas zu tun. Vielmehr geht dies nur über eine „Indirektion“. Ein Agent ist mit





einem Ziel ausgestattet: er handelt also *zielorientiert*. Im vorliegenden Fall mag das Ziel sein, Ihnen so gut wie möglich zu dienen. Sie können hoffen, dass der Agent Ihre Not erkennt und Ihnen den gewünschten Nachschub besorgt – vor allem, wenn sie deutlich und leidend auf Ihren Durst und Hunger hingewiesen haben. Vielleicht kommt er aber auch zu dem Schluss, dass er Ihrem Wohlbefinden besser dient, wenn er die von Ihrem Fußballfieber arg angefressene Ehefrau auf bessere Gedanken bringt – und dann ist Essig mit Bier und Chips. Das Beispiel erlaubt somit eine erste Annäherung an eine Definition für Softwareagenten:

sie relevante Änderungen in der Realwelt wahr und versuchen pro-aktiv ihre Umgebung so zu verändern, dass sie besser zu ihren Zielen passt. Dabei unterscheidet man zwischen zwei grundsätzlichen Verhaltensweisen, die sich im Grade ihrer Intelligenz und Anpassungsfähigkeit sowie der Reaktionsgeschwindigkeit unterscheiden.

Die weniger komplexe (aber durchaus nicht weniger leistungsfähige) Variante ist der *reaktive* Agent. Er hat entweder nur ein extrem einfaches oder gar kein Verständnis der Realwelt, kann also seine Umgebung und deren Veränderungen nicht wirklich interpretieren. Solche

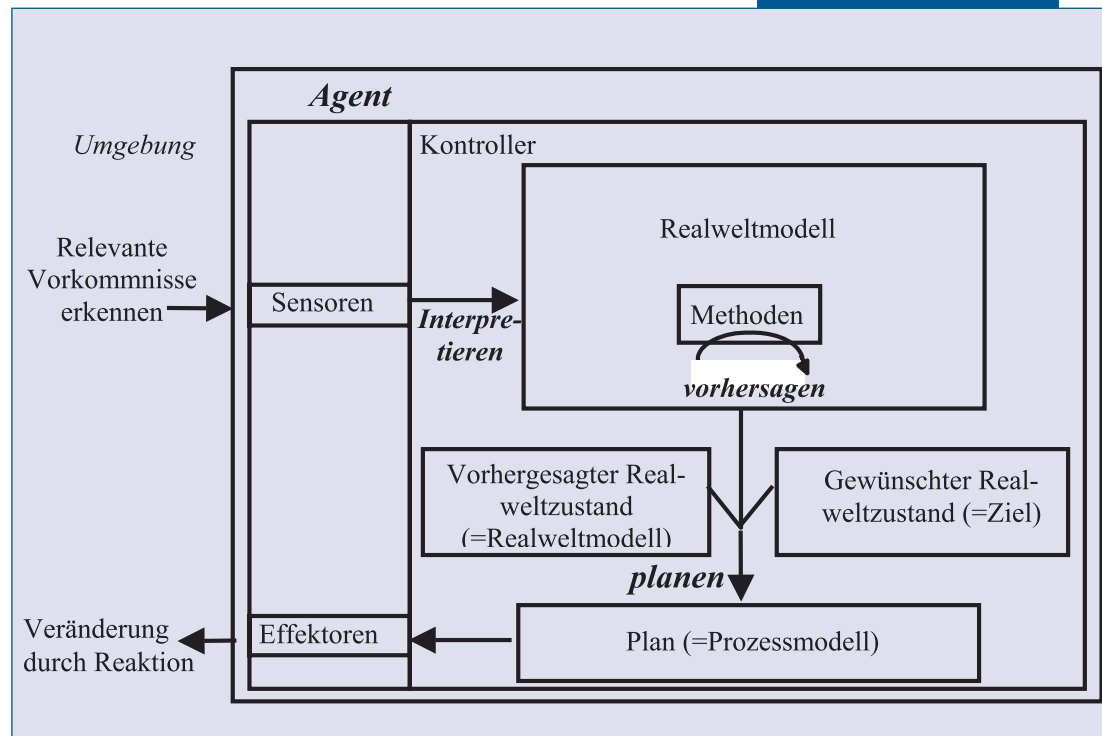
Ein *Softwareagent*, auch *autonom*er bzw. *intelligenter Agent* genannt, ist ein Computerprogramm, das *zielorientiert* und *autonom* in einem dynamischen Umfeld für einen menschlichen oder maschinellen Auftraggeber arbeitet. Der Agent weist in der Art, wie er seine Ziele über das Ausführen der ihm eigenen Methoden und Werkzeuge erreicht, einen gewissen Grad an *Flexibilität*, *Kreativität* und *Intelligenz* auf.

Annahmen, Wünsche, Absichten

Auch knapp 20 Jahre nach der ersten Verwendung des Begriffs Softwareagent in der Informatik ist dieser immer noch nicht eindeutig definiert. Stattdessen existiert eine Reihe von Definitionen, von denen keine als dominierend angesehen werden kann. Fast immer wird jedoch unterstellt, dass Agenten die bereits in der ersten Annäherungsdefinition aufgeführten Fähigkeiten besitzen.

In der weit verbreiteten so genannten BDI-Architektur verfügen Agenten über Annahmen (*beliefs*), Wünsche (*desires*) und Absichten (*intentions*). *Annahmen* beschreiben den Teil der Welt, den der Agent verstehen kann – allerdings aus seiner subjektiven Sicht. *Wünsche* beschreiben die Ziele, die der Agent gerne erreichen möchte. Unter *Absichten* sind die Pläne zu verstehen, die ein Agent ausführen will, um in Abhängigkeit von seinem Verständnis der Realwelt sein vorgegebenes Ziel zu erreichen.

Die bisherigen Diskussionen implizieren, dass Agenten *aktiv* bzw. *agierend* sind: sie nehmen für



Agenten arbeiten oft musterbasiert. Sie besitzen einen Musterspeicher. Immer, wenn in der Realwelt eines dieser Muster auftaucht, wird die dazugehörige (Re-)Aktion ausgelöst. Dies erlaubt eine schnelle, aber eher statische Reaktion auf Umweltveränderungen. Eine Lernfähigkeit ist kaum gegeben.

Deliberative Agenten können Umweltveränderungen wegen der ihnen innewohnenden subjektiven Beschreibung ihrer Umwelt interpretieren. Als Reaktion auf für sie relevante Ereignisse entwickeln sie dynamisch einen *Plan*, wie die Realwelt im Sinne des Agenten möglichst optimal beeinflusst werden kann (Abbildung 1). Solche Agenten sind in ihren Reaktionen flexibel und anpassungsfähig. Schnell sind sie nicht, denn der enorme Berechnungsaufwand zur Interpretation der Realwelt und zur Erstellung eines Reaktionsplanes kostet Zeit.

Abbildung 1: Funktionsweise eines pro-aktiven und deliberativen Agenten

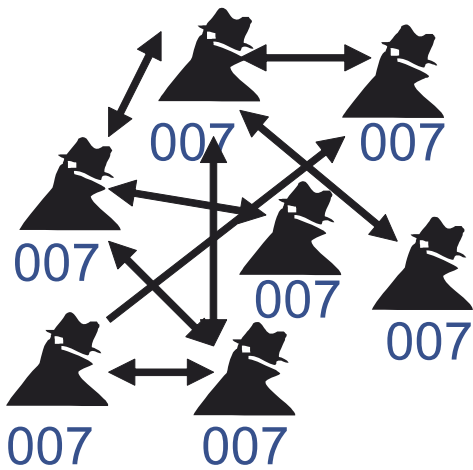


Abbildung 2:
Multiagentensystem (MAS)

Ein *hybrider Agent* stellt eine Kombination aus reaktivem und deliberativem Agenten dar. Er versucht zunächst, ein bereits vordefiniertes Verhaltensmuster auszuführen, arbeitet parallel dazu aber auch an genauen Interpretationen der Realwelt, um gegebenenfalls eine spezifischere und angepasste Reaktion nachzuschieben.

Trotz aller Autonomie – auch ein James Bond arbeitet selten allein. Zum Erfolg braucht er sein Netzwerk: Den knochentrockenen Chef M, den schrulligen Waffenmeister Q, die spröde Miss Moneypenny, den drahtigen CIA-Kollegen Felix und nicht zuletzt die ihn anziehenden schönen Frauen. Im Cyberspace ist das ganz ähnlich. Will die Agententechnologie erfolgreich sein, ist es unumgänglich, das Konzept eines Softwareagenten auf eine Menge (oder Gemeinschaft) von Softwareagenten zu erweitern.

007 & Co. – vom Einzelkämpfer zum Multiagenten

Ein Multiagentensystem stellt eine Gemeinschaft von Agenten dar, die aufgabenbezogen miteinander kooperieren, um Lösungen für Probleme zu erarbeiten, die ein einzelner Softwareagent nicht lösen kann. In unserer Realwelt kann man sich darunter eine Menge von internationalen Experten vorstellen, die gemeinsam und kooperativ an der Lösung eines Problems arbeiten. Und wie in der Realwelt gibt es auch im Cyberspace die üblichen Hindernisse.

Zunächst muss eine gemeinsame Kommunikationsplattform geschaffen werden, damit sich die Experten überhaupt verständigen können. Dies bedeutet, dass entweder ständig zwischen den Muttersprachen der verschiedenen Experten hin- und herübersetzt wird oder dass eine Übersetzung in eine neutrale Zwischensprache stattfindet, von der dann jeweils in die Zielsprache übersetzt wird. Die letzte Lösung ist vor allem dann sinnvoll, wenn innerhalb eines Expertenteams eine größere Zahl von unterschiedlichen



Muttersprachen gesprochen wird. In der Realwelt einigt man sich dann häufig auf Englisch als Kommunikationsbasis.

Problemlösung im Team

Eine lose gekoppelte, also jederzeit dynamisch änderbare Menge von autonomen Agenten, die bedarfsbezogen und auf eigenen Willen hin zusammenarbeitet, um komplexe Probleme zu lösen, nennt man ein *Multiagentensystem* (MAS) (Abbildung 2). Ähnlich wie in einer Arbeitsgruppe von Menschen hängt auch der Erfolg kooperierender Agenten von ihrer Verfügbarkeit, ihren Zielen und Kompetenzen sowie ihrer Bereitschaft und Fähigkeit zum Teamwork ab. Durch MAS erarbeitete Lösungen sind nicht notwendigerweise optimal. Auch ist nicht sichergestellt, dass Agentensysteme überhaupt eine Lösung zu einem vorgegebenen Problem finden – selbst dann nicht, wenn klar ist, dass es eine solche gibt. Insofern ähneln MAS der menschlichen Gemeinschaft. Innerhalb eines MAS gibt es gute und schlechte, fähige und unfähige, gründliche und lasche, hilfsbereite und egoistische Agenten. Auch können schwarze Schafe ihr Unwesen treiben.

Die Bearbeitung einer komplexen Aufgabe beginnt mit ihrer Übermittlung an einen Schnittstellenagenten (*interface agent*). Nach Erfassung der Problemstellung wird der Problemlösungsprozess eingeleitet. Er besteht aus drei Phasen. Zunächst wird das Gesamtproblem rekursiv in kleinere Problemeinheiten zerlegt, an deren Lösung sich einzelne Agenten heranwagen. Da ein Agent autonom ist, besteht kein Einfluss darauf, ob er sich an einem gemeinsamen Projekt beteiligt und wenn ja, wie er die ihm zugeordneten Arbeiten erledigt. Ein Agent mag die gesamte von ihm akzeptierte Aufgabenstellung selbst lösen oder Teilaufgaben an andere Agenten zur Bearbeitung weiterreichen (Abbildung 3). Die zweite Phase besteht aus der *Bearbeitung* und *Lösung* der jeweiligen Probleme. Hier kommt die individuelle Expertise des Agenten zum Tragen. Immer dann, wenn die Beziehungen zwischen den Teilproblemen nicht verloren gehen dürfen und deshalb Abstimmungsprozesse stattfinden müssen, sind aber auch seine kooperativen Fähigkeiten gefragt.

In der letzten Phase werden die einzelnen Teillösungen zur Gesamtlösung zusammengesetzt (*Komposition*). Die Phasen müssen nicht streng nacheinander ablaufen, sondern können ineinander verwoben sein, wobei auch ein Rücksetzen möglich ist, falls eine Sackgasse betreten wurde.

Unser ursprüngliches „Bier und Chips“-Beispiel erfordert nur eine einfache Form der Kooperation. „Unser“ Agent definiert in der Dekompositionsphase die Teilprobleme „Bier besorgen“ und „Chips besorgen“ und beauftragt damit jeweils geeignete kooperationswillige Agenten. In der Lösungsphase besorgen die Agenten die Sachen. Sollte der ursprünglich ausgewählte Kiosk keine Chips haben, kann ein anderer Agent (im Zuge des Backtracking) diese am Bahnhof besorgen.

In der Kompositionsphase nimmt unser Agent Bier und Chips in Empfang und liefert beides bei Ihnen ab. Die Autonomie impliziert übrigens, dass in einem MAS alle Agenten auf der gleichen Ebene angesiedelt sind, es also *keine Hierarchie* gibt, weshalb eine Zusammenarbeit immer wieder individuell ausgehandelt werden muss.

Kommunikation ist alles

Die Basis einer jeden Kooperation oder Koordination ist die Kommunikation. Sie geschieht auch bei Agenten über eine Sprache, die so genannte *Kommunikationssprache*. Im Gegensatz zur natürlichen Sprache, wo Syntax und Semantik eher eine Einheit bilden, wird bei der Agentenkommunikation zwischen diesen beiden Ebenen sauber unterschieden. Die *Kommunikationssprache* repräsentiert die *Syntax* als Grundlage der Kommunikation. Die Kommunikation ist allerdings auf wenige so genannte *Sprechakte* begrenzt. Typische *Sprechakte* sind die Lieferung von oder Nachfrage nach Informationen, die Anfrage nach und das Bestätigen oder Ablehnen von Zusammenarbeit oder das Akzeptieren oder Zurückweisen eines Vorschlages. Jeder Sprechakt basiert auf einem exakt vorgegebenen Muster. Diese engen Vorgaben erlauben eine eindeutige Interpretation des Sprechaktes und erfüllen damit die Grundvoraussetzung für eine maschinelle Kommunikation. Zusätzlich liefert ein Sprechakt auch alle notwendigen Verwaltungs- und Interpretationsinformationen wie den Absender, den Empfänger, Aussagen, worauf sich dieser Sprechakt bezieht, die verwendete Ontologie, die Sprache, in der dieser formuliert ist und den eigentlichen Inhalt des Sprechaktes. Letzteres repräsentiert als eigentliche Aussage des Kom-

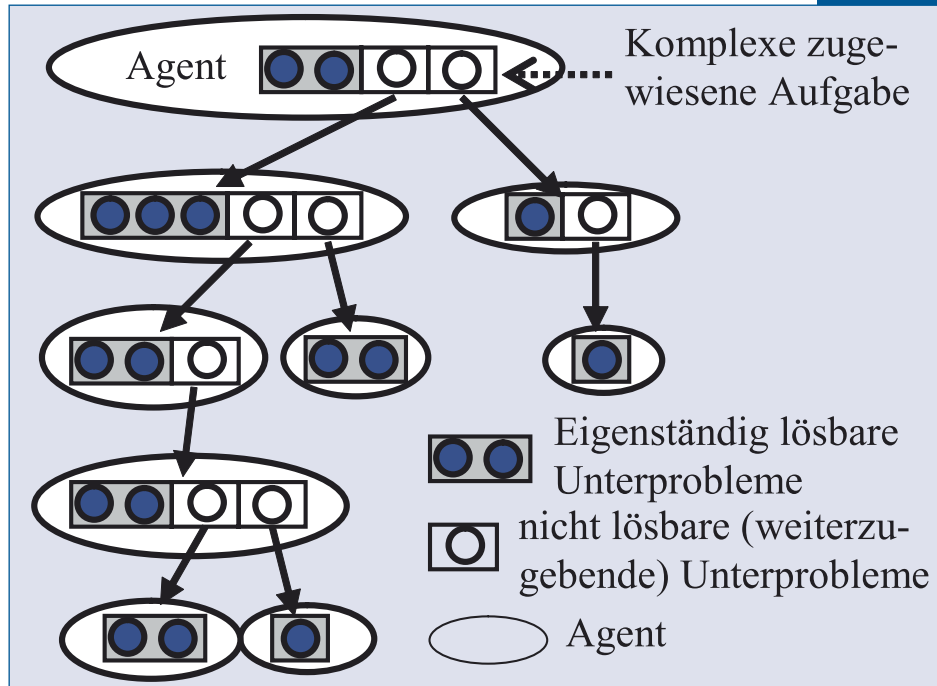


Abbildung 3:

Schrittweise Zerlegung eines komplexen Problems

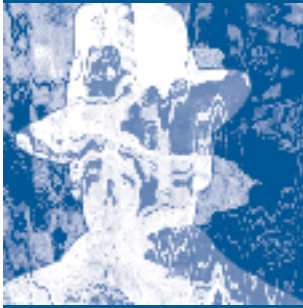
munikationsvorganges dessen *Semantik*. Das Ganze lässt sich gut mit einer Kommunikation über einen Brief vergleichen. Der Briefumschlag mit seinen Informationen über Absender und Empfänger repräsentiert die weltweit eindeutig interpretierbare Kommunikationssprache, während der eigentliche Inhalt nur für ausgewählte Adressaten interpretierbar sein kann.

Gewusst wie

Ein Agent muss eine gewisse Form von *Intelligenz* aufweisen, um auf seinem fest umrissenen Gebiet Aufgabenstellungen zu bearbeiten. Dies impliziert, dass er genügend Allgemeinwissen besitzen muss, um zu erkennen, ob sein Problemlösungspotenzial von der Außenwelt benötigt wird bzw. ob in der Außenwelt Veränderungen vor sich gehen, die in Bezug auf seine ureigenen Interessen und Ziele von Wichtigkeit sind. Schließlich muss er auch die Fähigkeit haben, sich im Alltag zu behaupten – beispielsweise bei der Einwerbung und organisatorischen Abwicklung von Aufträgen.

Im Vergleich zu Expertensystemen sind einzelne Softwareagenten eher Problemlöser für eng eingegrenzte Probleme. Die Lösung komplexerer Aufgaben erfordert daher die *Zusammenarbeit* von Agenten. Und für eine erfolgreiche Zusammenarbeit ist es essentiell, dass man sich gegenseitig versteht. Beim Menschen ist Verständigung bei Problemen vor allem dadurch möglich, dass jeder Mensch über einen Satz an Allgemeinwissen (Weltwissen) verfügt. Dies erlaubt ihm, Informationen einzuordnen, zu verarbeiten und Schlüsse zu ziehen. Agenten fehlt dieses breite Weltwissen. Ihnen kann bestenfalls ein eng eingeschränktes Wissen über ihr Aufgabengebiet und





über für sie wesentliche Aspekte ihrer Umwelt mitgegeben werden.

Wenn nun unter diesen Voraussetzungen Agenten miteinander kooperieren wollen, muss sichergestellt sein, dass sich ihr Wissen ausreichend überlappt und dass vor allem auch das Verständnis der Terminologie und der Zusammenhänge dieses überlappenden Themengebiets dasselbe ist. Wenn zwei Agenten über dasselbe reden, müssen sie dasselbe meinen. Dazu dienen *Ontologien*. In unserem Zusammenhang kann man sich diese als ein deutlich erweitertes Nachschlagewerk vorstellen, in dem die Begriffe eines Fachgebietes eindeutig, aussagekräftig und in maschinenlesbarer Form beschrieben sind. Ontologien beschreiben Domänen, wobei deren Spektrum je nach Nutzungsziel von sehr feingranular (spezifisch) bis sehr grobgranular (abstrakt, übergreifend) reichen kann. Man spricht auch von expliziter *Konzeptualisierung* von Phänomenen der Realwelt. Eine Konzeptualisierung ist eine abstrahierte, aber vollständige und häufig auch formalisierte Sichtweise auf einen Realweltausschnitt (bzw. auf eine Domäne). Eine Ontologie besteht dann aus einem *Terminorrat*, in dem alle Begriffe, die eine Konzeptualisierung beschreiben sollen, zusammen mit allen syntaktisch zulässigen Begriffsverknüpfungen aufgeführt sind.

Der eigentlich interessante Aspekt einer Ontologie ist die Beschreibung ihrer *Semantik*. Dies geschieht häufig über Regeln, über die Begriffe miteinander in Beziehung gesetzt und erklärt werden. Ontologien sind nur ein erster Schritt zur semantischen Beschreibung einer Domäne, da sie nur eine stark eingeschränkte Beschreibung der Semantik der Begriffe einer Domäne erlauben. Zudem bedeutet eine Überlappung von Ontologien nicht, dass die den Ontologien zu Grunde liegende Konzeptualisierung und Semantik konfliktfrei und widerspruchsfrei ist. Hier ist noch viel Arbeit erforderlich.

Agentenplattformen

Eine zentrale Rolle bei der Einführung und Durchsetzung von Konzepten und Techniken spielt die Standardisierung. In Bezug auf Softwareagenten und Multiagentensysteme hat sich die so genannte *Foundation of Intelligent Physical Agents (FIPA)* sehr hervor getan. Sie widmet sich der Standardisierung auf allen Ebenen der Agentenentwicklung. Dazu gehört auch, dass die FIPA Vorgaben gemacht hat, auf deren Basis so genannte Agentenplattformen entwickelt werden können. Die vielleicht bekannteste frei verfügbare Agentenplattform ist JADE (*Java Agent DEvelopment Framework*). JADE kann als ein

in Java implementierter Softwarebaukasten zum Entwurf eines Agentenstaates angesehen werden. In dem Staat gibt es Bürger (die Agenten), die alle eine gemeinsame Sprache sprechen (vergleichbar mit einem individuellen Land wie Deutschland). Die Agentenplattform offeriert einen Lebenszyklusservice zum Einrichten, Löschen und Warten von individuellen Agenten sowie der Agentenplattform selbst, weiße (Name und Kontaktdaten von Softwareagent) und gelbe (Verzeichnis von Kontaktdaten, geordnet nach Service- und Produktkategorien) Seiten sowie den allgemeinen Service der Zustellung von Mitteilungen. Dabei bleibt sowohl die eigentliche Zustellung als auch der eigentliche Aufenthaltsort eines Agenten innerhalb eines Staates transparent, was bedeutet, dass die Zustellung automatisch vom System vorgenommen wird, ohne dass bekannt sein muss, wo ein Agent sich gerade befindet. Als optionale Services werden noch eine Mensch-Maschineschnittstelle, ein Ontologieservice und ein Service zur plattformübergreifenden Interaktion zwischen Agenten angeboten. Letzterer ermöglicht Agenten aus verschiedenen Plattformen (und damit auch aus verschiedenen Welten mit unterschiedlichen Muttersprachen) miteinander zu kommunizieren. Eine Agentenplattform legt das Rahmengerüst einer „internationalen“ Gemeinschaft fest inklusive der angebotenen Kommunikationswege und Informationsquellen, also die generelle Struktur und Funktionsweise der Gemeinschaft (insbesondere ihren Aufbau und ihr administratives Grundgerüst). Die zur Lösung der eigentlichen Aufgaben der Gemeinschaft notwendige Intelligenz kann auf dieser Ebene nicht modelliert werden, sondern muss explizit programmiert werden. Schließlich gibt es noch einen Service zur plattformübergreifenden Interaktion zwischen Agenten. Er erlaubt die Kooperation zwischen Agenten aus verschiedenen Plattformen (und damit auch aus verschiedenen Staaten mit unterschiedlichen Muttersprachen z. B. die Staaten in Europa) und bietet alle dazu notwendigen Services an (wie die Übersetzung von einer Sprache in eine andere).

Alter Wein in neuen Schläuchen?

Was unterscheidet die Agententechnologie von bestehenden Expertensystemen oder objektorientierten Technologien? Einem Softwareagenten wird ebenso wie einem Expertensystem ein gewisser Grad an Intelligenz unterstellt. Allerdings fällt das einem Agenten zugeordnete Wissen oft eher klein aus – Agenten sind eher enge Fachspezialisten, die sich zwecks Lösung komplexerer Aufgaben mit anderen Agenten zusammenschließen. Expertensysteme können durchaus vergleichsweise breites Wissen besitzen. Dafür

CCE Congress Center Essen: Ein Ereignis für sich.

Hier die komplette Checkliste,
um Ihren perfekten Kongress
zu organisieren:



CCE anrufen:
0201.72 44-878

Sie kennen zahlreiche Veranstaltungsagenturen. Aber kennen Sie auch eine, die in den letzten Jahren von ihren Kunden nur Bestnoten bekommen hat? Dann ist es Zeit, den Full Service des CCE kennenzulernen.

+49.(0)201.72 44-878
www.cc-essen.de



CCE – Ein Geschäftsbereich der Messe Essen GmbH



haben Agenten Ziele, sind reaktiv und häufig auch proaktiv und können mit ihrer Umwelt kommunizieren – sind also keine passive Wissensquelle, auf die nur bei Bedarf von außen zugegriffen wird.

Auch gegenüber Objekten gelten diese Unterscheidungsmerkmale. Zusätzlich ist der Grad der Autonomie ein anderer, da Agenten die Entscheidungsgewalt in ihren Händen haben, während Objekte reagieren müssen, sobald sie über einen entsprechenden Methodenaufruf aktiviert wurden. Es gibt deshalb nicht wenige, die in der agentenbasierten Softwareentwicklung den Nachfolger der objektorientierten Softwareentwicklung sehen. Wie bei der Objektorientierung erlaubt auch die Agentenbasierung eine starke Modularisierung der Anwendung. Während bei der Objektorientierung allerdings der Datenfluss, die Methodenausführung und die Objektaktivierung explizit gesteuert und vordefiniert werden müssen, können bei der agentenbasierten Softwareentwicklung diese Aufgaben auf die Agenten abgeschoben wer-

den. Auch die Fehlerbehandlung wird deutlich erleichtert, da durch den fehlenden starren Programmablauf jederzeit andere Agenten eingespannt werden können, falls es Probleme bei einer Aufgabenbewältigung gibt. Viele Pluspunkte also für James Bonds elektronische Kollegen, deren Potenzial und Fähigkeiten im Lauf der Entwicklung mit Sicherheit noch steigerbar sein werden.

Kontakt

Prof. Dr. Rainer Unland

Datenverwaltungssysteme und
Wissensrepräsentation

Tel.: 02 01 / 1 83 - 34 21

Fax.: 02 01 / 1 83 - 44 60

IP-Tel. Skype: unlandr

Rainer.Unland@icb.uni-due.de

<http://www.cs.uni-essen.de/dawis/>

