

6 Wissensbasierte Systeme und Mensch-Computer-Kommunikation

6.1 Einleitung

Bei der Entwicklung von Computersystemen treten häufig technische Detailprobleme in den Vordergrund der Betrachtungen. Wir müssen darauf achten, daß wir die folgenden Ziele dabei nicht aus den Augen verlieren.

- Computersysteme sollen *dem Menschen mehr Arbeiten abnehmen*, insbesondere wenn es sich um Arbeiten handelt, die das menschliche Denkvermögen nicht ohne weiteres zu leisten vermag. Ferner sollen sie Routinearbeiten, die sehr zeitaufwendig sind, aber wenig Kreativität erfordern, und Arbeiten in gefährlichen Umgebungen übernehmen.
- Computersysteme sollen *für mehr Menschen zugänglich* gemacht werden. Es sollte keine mehrjährige Ausbildung erforderlich sein, um ein Computersystem auf sinnvolle Art und Weise benutzen zu können.

Diese Ziele zu erreichen, erfordert primär nicht grundlegend neue Hardware, bessere Betriebssysteme oder neue Programmiersprachen (auf diesen Gebieten sind Fortschritte leichter zu erreichen), sondern Forschungen im Bereich der *Wissensbasierten Systeme (WBS)* und der *Mensch-Computer-Kommunikation (MCK)*. Für diese Forschungsarbeiten müssen geeignete Werkzeuge entwickelt und in prototypischen Anwendungen erprobt werden.

6.2 Wünschenswerte Eigenschaften zukünftiger Computersysteme

6.2.1 Menschenorientierte Systeme

Wissenschaftlicher und technischer Fortschritt haben in den letzten zwanzig Jahren dazu geführt, daß die Kommunikationsfähigkeiten des Computers schrittweise verbessert worden sind. Diese Entwicklung hat damit begonnen, daß Compiler es ermöglicht haben, Programme in höheren Programmiersprachen zu schreiben – eine erste Aufhebung des Zwangs, daß der Mensch sich dem Computer anpassen muß. Diese Entwicklungsrichtung läßt sich auch charakterisieren als

ein Fortschritt vom „wie“ zum „was“. Anstatt der Maschine explizit alle Einzelschritte mitzuteilen, soll es möglich sein, das Problem anzugeben, und der Computer benutzt sein Wissen, um zu einer Lösung zu kommen.

Der Anspruch, menschenorientierte Systeme zu entwickeln, impliziert, daß die Entwicklungsrichtung bei Computersystemen umgekehrt werden muß. Die traditionelle Vorgehensweise, Systeme von „innen nach außen“ zu entwickeln (d. h. man geht aus von der Hardware, entwirft Software dafür, fügt als letztes noch eine Benutzerschnittstelle hinzu), muß ersetzt werden durch einen Prozeß, in dem Systeme von „außen nach innen“ entwickelt werden (d. h. am Anfang wird das sozio-technische System charakterisiert, in dem das zu entwickelnde Computersystem einzubetten ist, und dafür wird anschließend die geeignete Soft- und Hardware geschaffen).

In der Vergangenheit haben technologische Entwicklung vorwiegend dazu beigetragen, die motorischen und sensorischen Eigenschaften des Menschen zu unterstützen (z. B. Hammer, Mikroskop). Die Herausforderung an die Informationstechnik besteht darin, die kognitiven Fähigkeiten zu erweitern, d. h. Computer sollen als Verstärker der menschlichen Intelligenz dienen.

6.2.2 Symbiotische Systeme

Symbiotische Systeme basieren auf einer Ergänzung menschlicher Fertigkeiten durch den Computer mit dem Ziel, gemeinsam eine Aufgabe auszuführen, die weder Mensch noch Computer allein ausführen können. In vielen Bereichen haben die Probleme eine derartige Komplexität erreicht, daß der Mensch ohne maschinelle Unterstützung nicht mehr zurechtkommt.

Untersuchungen (z. B. bei Piloten und bei Beschäftigten in Kernkraftwerken) haben gezeigt, daß die Leistungsfähigkeit des Menschen erheblich beeinträchtigt ist, wenn er in eine passive Rolle gedrängt wird. Konstante Unterforderung verhindert, daß er seine Tätigkeit in jeder Phase geistig verfolgt und gegebenenfalls handelnd eingreift. Symbiotische Systeme stehen im Gegensatz zu Systemen, in denen der Mensch zum Lückenbüßer in einem automatischen, sich selbst steuernden und kontrollierenden System gemacht wird oder sich mit geistiger Fließbandarbeit begnügen muß.

Bei symbiotischen Systemen ist die Kommunikation der beiden Partner von zentraler Bedeutung, weil diese *gemeinsam* eine Aufgabe erledigen. Wir sehen für eine Aufgabenteilung zwischen Mensch und Maschine die folgenden Schwerpunkte:

1. Der Mensch:

- liefert die Ziele
- benutzt umfangreiches Allgemeinwissen
- definiert die Teilprobleme und ihre Beziehungen zueinander
- baut auf vorhergehenden Erfahrungen auf
- integriert Wissen aus verschiedenen Bereichen
- löst Probleme durch Analogieschlüsse
- selektiert geeignete Repräsentationen
- kontrolliert Teillösungen

2. Die Maschine:

- übernimmt die Funktion einer externen Gedächtnishilfe (um umfangreiche Informationsmengen zu speichern und Zwischenergebnisse festzuhalten)
- liefert Abstraktionsebenen (dadurch, daß Prozeduren, Pakete und Moduln definiert werden können)
- kontrolliert die Auswirkungen von Veränderungen und versucht, Inkonsistenzen zu vermeiden oder aufzudecken
- veranschaulicht die Konsequenzen unserer Annahmen (ermöglicht ein Vorgehen entsprechend der von Popper postulierten Methode der kritischen Irrtumsbeseitigung¹¹⁾)
- zerlegt komplexe Informationsstrukturen in verschiedene Perspektiven und verbirgt irrelevante Einzelheiten
- lenkt unsere Aufmerksamkeit auf wichtige Ereignisse (z. B. durch visuelle Reize, wie verschiedene Farben, „Reverse Video“, Blinken, usw.)
- ermöglicht, daß die Konsequenzen von Aktionen rückgängig gemacht werden können („UNDO-Befehl“), und fördert damit exploratives und kreatives Verhalten
- erzeugt dynamisches Verhalten entsprechend statischer Beschreibungen.

6.2.3 Konviviale Systeme

Der Begriff des „convivial tool“ (auf deutsch: „Konviviales Werkzeug“) wurde von Ivan Illich geprägt. Die beste Definition findet man in der englischen Ausgabe⁸⁾:

„Tools are intrinsic to social relationships. An individual relates himself in action to his society through the use of tools which he actively masters, or by which he is passively acted upon. To the degree that he masters his tools, he can invest the world with his meaning; to the degree that he is mastered by his tools, the shape of the tool determines his own self-image. Convivial tools are those which give each person who uses them the greatest opportunity to enrich the environment with the fruits of his or her vision. Tools foster conviviality to the

extent to which they can be easily used, by anybody, as often or as seldom as desired, for the accomplishment of a purpose chosen by the user."

Eines der Hauptanliegen von Illich ist: Der Einzelne wird mehr und mehr des Bewußtseins beraubt, er könne seine Umwelt aktiv gestalten. Diese Angst vor dem Beherrschtwerden ist sicher einer der wesentlichen Beweggründe für die derzeitige Technologie- und Computerfeindlichkeit. Illich setzt sich in diesem Zusammenhang für eine „Entzunftung der Wissenschaft“ ein: Diese sollte sich bemühen, die Werkzeuge zu vereinfachen, um damit jedem die Möglichkeit zu bieten, seine unmittelbare Umwelt zu formen.

Werkzeuge sind konvivial, wenn sie „innerhalb des jeweiligen Arbeits- und Lebenszusammenhangs, in Funktionalität und Struktur angeeignet, d. h. denkend durchdrungen und handelnd bewältigt werden können“. Eine Anpassung ohne geistige Durchdringung, verbunden mit einer Anhäufung von Wissen über sachgerechte Benutzung und Bedienung von halbverstandenen Elementen, ist eine der großen Gefahren, die wir vermeiden müssen.

Die derzeitige Situation im Bereich der Computernutzung kann durch die Aussage „*Software is not soft*“ beschrieben werden. Trifft man aus Benutzersicht die folgenden Definitionen:

- *Software*: alle Teile eines Computersystems, auf die der Benutzer gestaltend eingreifen kann,
- *Hardware*: alle Teile, die, einmal vorhanden, nicht mehr modifiziert werden können,

so stellen sich heutige Computersysteme dem Benutzer fast ausschließlich als Hardware dar. In unserer Forschungsarbeit befassen wir uns schwerpunktmäßig damit, bei Computersystemen den Anteil der *echten Software* zu erhöhen und dem Benutzer die Umgestaltung des Systems nach seinen Anforderungen zu ermöglichen.

Im Gegensatz zu den meisten anderen Informationstechnologien (z. B. Fernsehen, Bildschirmtext, Videodiscs usw.), die *passiv* sind (d. h. der Benutzer kann nicht aktiv in deren Gestaltung eingreifen, allenfalls selektiv), kann der Computer dem Benutzer gestatten, eine *aktive, gestalterische Rolle* einzunehmen. Diese Modifikation durch den Endbenutzer ist für *offene Systeme* (als ein solches muß z. B. ein Büro betrachtet werden) von grundlegender Bedeutung, denn in ihnen können zu keinem Zeitpunkt alle möglichen Aktionen und Zustände vorhergesehen werden. Deshalb sind traditionelle EDV-Systeme, die nur genau spezifizierte Arbeitsabläufe durchführen oder unterstützen können, für zukünftige Entwicklungen nur von untergeordneter Bedeutung.

Einigen Mißverständnissen soll in diesem Zusammenhang vorgebeugt werden:

- Es ist nicht unbedingt zu fordern, daß die Aneignung von Wissen zur Handhabung solcher Systeme im „Handumdrehen“ stattzufinden hat; es kann durchaus erforderlich und akzeptabel sein, daß die meisterhafte Beherrschung ein langandauerndes und überlegtes Umgehen mit solchen Werkzeugen erfordert (für ein Auto muß auch ein Führerschein erworben werden).
- Die Entwicklung in Richtung wissensbasierter Systeme ist so verlaufen, daß das „*wie*“ durch das „*was*“ ersetzt worden ist, d. h. der Mensch hat zunehmend mehr Kontrolle an den Rechner abgegeben. Dieses erscheint an der Oberfläche zunächst als eine anti-konviviale Entwicklungsrichtung. Dem Menschen lag jedoch nichts an dieser Kontrolle (wie z. B. auch ein automatisches Getriebe im Auto von vielen Leuten bevorzugt wird); er wollte nicht die Maschine verwalten (z. B. durch das Angeben von Speicheradressen), sondern sie als intelligentes Werkzeug zur Lösung seiner Aufgaben benutzen.

6.2.4 Ergonomische Systeme

Der Ansatz der traditionellen Ergonomie reicht für Fragestellungen, wie sie sich im Zusammenhang mit Informationstechnologien ergeben, nicht aus. Moderne Computersysteme sollen den Menschen beim Problemlösen, beim Entscheiden, beim Planen und bei anderen kognitiven Tätigkeiten unterstützen. Um im Rahmen der *Software-Ergonomie* diese intelligenten Werkzeuge gestalten, analysieren und bewerten zu können, müssen neben den Rahmenbedingungen menschlicher Wahrnehmung, menschlichen Denkens und Handelns auch Arbeitsinhalte, Arbeitsabläufe sowie das Arbeitsumfeld berücksichtigt werden.

Unsere Erfahrungen und Zielvorstellungen im Bereich der Software-Ergonomie werden im folgenden in Thesenform mit kurzen Erläuterungen zusammengefaßt:

These 1: Die begrenzte Ressource bei der menschlichen Informationsverarbeitung ist die Aufmerksamkeit und nicht die verfügbare Information. Moderne Kommunikationstechniken haben die dem Einzelnen zur Verfügung stehende Informationsmenge ständig vergrößert. Fast jedem Menschen steht heutzutage mehr Information (in Büchern, auf Filmen usw.) zur Verfügung, als sie in ihrem gesamten Leben verarbeiten können. Die wirklich begrenzenden Faktoren sind damit die Zeit und die Aufmerksamkeit, die für die Verarbeitung von umfangreichen Informationsmengen aufgebracht werden kann¹²⁾. Eine der wichtigsten Funktionen von Computersystemen ist deshalb das intelligente Anbieten, Zusammenfassen und die kontrollierbare Darstellung von Informationen.

These 2: Die durch die unterschiedlichen Strukturen des menschlichen Gedächtnisses gegebenen Randbedingungen müssen bei der Gestaltung von Computersystemen berücksichtigt werden.

Die begrenzte Kapazität des Kurzzeitgedächtnisses ist eine der Einschränkungen, die menschliches kognitives Verhalten am meisten kennzeichnet. „Recognition Memory“ und „Recall Memory“ sind zwei Gedächtnisstrukturen, die sich in ihren Zugriffsmechanismen unterscheiden. Bei Systemen mit Menüauswahl wird z. B. nur das „Recognition Memory“ beansprucht, während bei Systemen mit Kommandoingabe das „Recall Memory“ beansprucht wird.

These 3: Der Fähigkeit zur effizienten Verarbeitung visueller Information durch den Menschen muß so weit wie möglich Rechnung getragen werden.

Moderne Techniken (z. B. Computergraphik, Farbdarstellungen, Fenster-Techniken usw.) eröffnen neue Möglichkeiten, diese Stärke der menschlichen Informationsverarbeitung besser zu nutzen.

These 4: Der Benutzer sollte in der Lage sein, sich ein adäquates Modell über die Funktionalität der ihm zur Verfügung stehenden Systeme zu schaffen.

Sofern der Benutzer nur an der für seine Tätigkeit relevanten Funktionalität eines Anwendungssystems interessiert ist, sollten tiefere Systemebenen (Betriebssystem, Programmiersprache, File-System) vor ihm verborgen bleiben.

These 5: Computersysteme müssen sich adaptiv gegenüber verschiedenen Benutzergruppen verhalten oder müssen sich durch den Endbenutzer adaptieren lassen.

„Den“ Benutzer eines Computersystems gibt es nicht. Jeder Benutzer hat einen speziellen Kenntnisstand in der Bedienung des Systems sowie spezielles Fachwissen. Er hat bestimmte Fähigkeiten in der Aufnahme visueller Information und der Ausführung motorischer Tätigkeiten, wie der Bedienung einer Tastatur oder eines Zeigeinstruments. Derartige spezielle Eigenschaften sind nicht unveränderlich und können damit nicht fest eingeplant werden. Wir benötigen Systeme, die sich an die speziellen Eigenschaften und Erfahrungen des jeweiligen Benutzers anpassen oder anpassen lassen. Solche *adaptiven* bzw. *adaptierbaren Systeme* sollen ihm insbesondere Möglichkeiten zur Gestaltung der Benutzerschnittstelle anbieten^{6, 2)}.

These 6: Bei der Bedienung mehrerer Anwendungssysteme durch einen Benutzer muß gewährleistet werden, daß diese in einer uniformen Art und Weise zu bedienen sind.

Die Konsistenz von Benutzerschnittstellen zu verschiedenen Anwendungssystemen ist eine Grundvoraussetzung für die sichere Bedienung von Systemen sowie den Transfer von Handlungswissen zwischen verschiedenen Tätigkeiten. Beispielsweise werden die Cursorbewegungen in einem bildschirmorientierten Editiersystem bei häufiger Benutzung nur noch unterbewußt kontrolliert (analog zur Bedienung des Schalthebels im Auto). Diese Handlungen werden beim Übergang auf ein anderes Anwendungssystem bewußt oder unbewußt auch dort ausgeführt. Um effizientes Handeln wirkungsvoll zu unterstützen, muß vermieden werden, daß der Benutzer häufig zwischen verschiedenen Konventionen hin- und herwechseln muß.

These 7: Die Benutzerschnittstelle eines Computersystems muß von Spezialisten gemäß den genannten Forderungen entwickelt werden, so wie das Anwendungssystem vom Anwendungsspezialisten konzipiert und vom Anwendungsprogrammierer realisiert werden soll.

Da die Benutzerschnittstelle die Qualität der Arbeit mit einem System entscheidend beeinflusst und sie darüber hinaus nicht für jede Anwendung neu konzipiert werden sollte (Konsistenz), sollte der Entwicklungsprozeß bei Systemen umgekehrt werden: Anstatt Systeme von innen nach außen zu entwickeln, sollte dieser Prozeß in die andere Richtung verlaufen, d. h. die Benutzerschnittstelle sollte als Einbettung für die Anwendung angesehen werden und nicht umgekehrt.

These 8: Die vom System ausgeführten Aktionen sollten auf Wunsch des Benutzers rückgängig gemacht werden können (UNDO).

Viele Tätigkeiten (z. B. Design, Planung, Konstruktion) erfordern *exploratives Arbeiten*, da der mögliche Lösungsraum praktisch unüberschaubar ist, und der Mensch erst durch kritische Begutachtung eines Resultats entscheiden kann, ob die ausgeführten Aktionen Schritte in die Richtung einer akzeptablen Lösung waren. Im Falle eines unbefriedigenden Ergebnisses soll der Benutzer die Möglichkeit haben, auf ein früheres Ergebnis mittels UNDO-Funktionen zurückzukehren.

6.2.5 „Intelligente“ Systeme

Die wichtigste Aufgabe der Forschungen zu wissensbasierten Systemen und zur Mensch-Computer-Kommunikation sollte sein, den Computer von einem undurchsichtigen, unverständlichen und unnahbaren Instrument in ein zuverlässiges und lebenserleichterndes Werkzeug zu verwandeln. Computersysteme müssen Wissen über sich selbst besitzen, mit dem sie ihr eigenes Handeln in ver-

ständiglicher Form erklären können. Es ist ein untragbarer Zustand, daß Computerprogramme Aussagen und Daten liefern, die als Grundlage für wichtige Entscheidungen herangezogen werden, obwohl in vielen Fällen keine Möglichkeit besteht, die Entstehungsgeschichte dieser Daten und Aussagen zu überprüfen. Wissensbasierte Systeme sind komplex, weil sie komplexe Zusammenhänge der realen Welt modellieren. Aus diesem Grunde ist es erforderlich, Mechanismen vorzusehen, um diese Komplexität zu verdecken und zu gestatten, die „Aussagen des Systems zur Rechenschaft“ zu ziehen.

Wie intelligent sollen die Systeme in Zukunft sein? Wenn nicht erhebliche Anstrengungen gemacht werden, einer breiten Bevölkerungsschicht Information und Zugang zu dieser neuen Technologie zu gewähren, dann wird das, was Wissenschaftler als Fortschritt empfinden, von Laien als Entfremdung und Beherrschung empfunden werden. „Computer zu unserer Unterstützung zu nutzen“ im Gegensatz zu „durch Computer verwaltet oder gar beherrscht werden“, sind zwei grundverschiedene Perspektiven für die Zukunft, die nicht in erster Linie von der Natur der Computer bestimmt werden, sondern vielmehr von der Art und Weise, wie wir sie benutzen.

Die Grenzen für mögliche Computeranwendungen können heute noch nicht eindeutig abgesteckt werden. Trotzdem ist zu erwarten, daß die Veränderungen durch die vielfältige Nutzung von Computern ähnlich tiefgreifend sein werden, wie die der industriellen Revolution. Mit den folgenden Entwicklungen werden wir uns intensiv auseinandersetzen haben:

1. *Veränderungen des Arbeitsmarktes im Bereich der Büroautomatisierung:* Die soziologischen Verschiebungen auf dem Arbeitsmarkt dürfen nicht auf dem Rücken von wenigen ausgetragen werden. Es sollte nicht übersehen werden, daß der Einsatz von Computern nur beschränkt von denen bestimmt werden wird, die Überlegungen anstellen, wie Computer zu lebenserleichternden Werkzeugen gestaltet werden können, sondern entscheidend durch eine am Gewinn orientierte Gesellschaftsordnung.
2. *Der Computer als Kontrollinstrument:* Die Problematik wird durch die Diskussion über Regelungen zum Datenschutz verdeutlicht. Nicht die technische Machbarkeit ist hier das kritische Merkmal, sondern die Ziele derjenigen, die Computer anwenden.
3. *Veränderungen von Arbeit:* Wie verhindern wir, daß der Mensch zu einem Lückenbüßer in weitgehend vollautomatischen Systemen wird und daß „geistige Fließbandarbeit“ die Tätigkeiten vieler Menschen dominiert? Mit unserer Projektarbeit¹⁾ wollen wir menschenorientierte Computersysteme schaffen und damit verhindern, daß Menschen zu computerorientierten Wesen degradiert werden.

6.3 Wissensbasierte Systeme und Mensch-Computer-Kommunikation

Wissensbasierte Systeme (WBS) und Mensch-Computer-Kommunikation (MCK) sind Querschnittstechnologien, d. h. sie sind für eine Vielzahl von Entwicklungen im informationstechnischen Bereich von Bedeutung. Forschungsarbeiten zur MCK können sich nicht nur auf die reine Kommunikationsform (z. B. Dialoge in natürlicher oder gesprochener Sprache im Gegensatz zu formalen Sprachen oder zu direkter Manipulation des Bildschirms) und die externe Informationsdarstellung auf dem Bildschirm beschränken. Ohne Wissen auf seiten des Systems kann nur sehr eingeschränkte Kommunikation stattfinden. Unser Forschungsansatz geht von dem empirisch nachweisbaren Zusammenhang aus, daß diese beiden Themen *untrennbar* zusammengehören⁷⁾. Dieser Zusammenhang wird durch folgende grundlegende Modellvorstellung in Bild 6.1 veranschaulicht.



Bild 6.1: Modell eines wissensbasierten Systems und seiner Benutzerschnittstelle

Ein menschlicher Experte verfügt neben seinem Expertenwissen in einem bestimmten Problembereich auch noch über andere Wissenskomponenten, die in obigem Bild angegeben sind. Wissen über Problemlösen, über Kommunikationsprozesse und über Kommunikationspartner ist für ein kooperatives Computersystem genauso wichtig wie für einen menschlichen Experten.

Bild 6.2 zeigt ein Modell eines traditionellen Systems, mit dem nur eine sehr eingeschränkte Kommunikation stattfinden kann, und das nur über wenig Wissen verfügt.

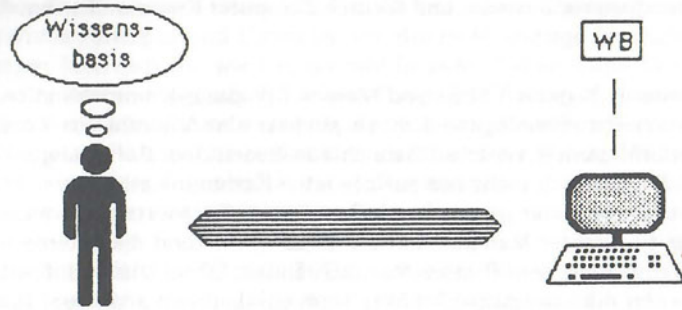


Bild 6.2: Traditionelles System

Gute MCK ohne umfangreiche Wissensbasis (Bild 6.3) hat folgende Analogie bei der menschlichen Kommunikation: Jemand kann gut reden, hat aber letztlich nicht viel zu sagen. Ein Beispiel dafür liefert das STAR-System von Xerox, das zwar eine schöne Benutzerschnittstelle hat, aber für bestimmte Aufgabstellungen (die Wissen über diese erfordern) bietet es nur unzureichende Unterstützung.



Bild 6.3: Gute MCK ohne umfangreiche Wissensbasis

Umfangreiches Wissen ohne Kommunikationsfähigkeiten (Bild 6.4) bedeutet für die menschliche Kommunikation: jemand, der viel weiß, aber dieses Wissen nicht weitergeben kann. Ein Beispiel dafür liefern die Expertensysteme, die in Stanford entwickelt worden sind, und mit denen nur über eine teletypeorientierte Schnittstelle kommuniziert werden kann.

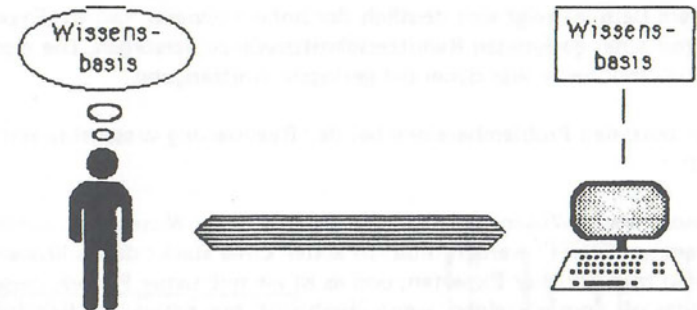


Bild 6.4: WBS ohne gute MCK

Es ist nicht anzustreben, daß jemand alles sagt, was er weiß, denn die begrenzte Ressource ist die menschliche Aufmerksamkeit und nicht die verfügbare Information. Mit den zwischengeschalteten Filtern (Bild 6.5) wird erreicht, daß Information benutzer- und aufgabenspezifisch angeboten wird.

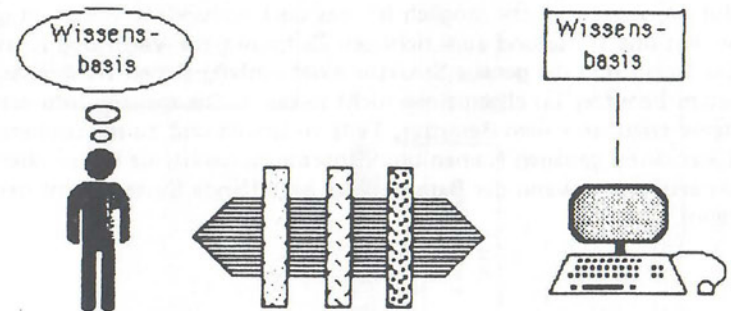


Bild 6.5: WBS und gute MCK

Eine empirische Evaluation des Dipmeterprojektes¹³⁾ ergab folgende Anteile bezüglich des Entwicklungsaufwandes:

- ```

* Inferenz-Maschine 8 % *
* Wissensbasis 22 % *
* Feature-Erkennung 13 % *
* Benutzerschnittstelle 42 % *
* Produktionsumgebung 15 % *
* *

```



Bei diesem Beispiel zeigt sich deutlich der hohe Aufwand, um ein Expertensystem mit einer geeigneten Benutzerschnittstelle zu versorgen. Die eigentliche Inferenz-Komponente war dabei die geringste Anstrengung.

Die drei zentralen Problembereiche bei der Realisierung wissensbasierter Systeme sind:

1. **Wissenserwerb:** Wissensbasierte Systeme benötigen Wissen, das zunächst ins System „gefüttert“ werden muß. In erster Linie steckt dieses Wissen in den Köpfen menschlicher Experten, und es ist ein mühsamer Prozeß, dieses Wissen (das oft vom einzelnen, wenn überhaupt, nur schwer explizit formuliert werden kann) auf die Maschine zu übertragen.
2. **Wissensrepräsentation:** In den vergangenen Jahren wurde eine Vielzahl von Mechanismen zur Wissensrepräsentation entwickelt. Unsere eigene Entwicklung ObjTalk<sup>9)</sup> ist eine objektorientierte Sprache, in der Wissen mit Hilfe von Klassen und Instanzen, die in ein Vererbungsnetzwerk eingebunden sind, repräsentiert wird.
3. **Wissensnutzung:** Ein mit viel Wissen ausgestattetes System ist von geringem Nutzen, wenn es nicht möglich ist, das dort vorhandene Wissen in geeigneter Art und Weise und zum richtigen Zeitpunkt zur Verfügung zu stellen. Der Inhalt und die genaue Struktur einer umfangreichen Wissensbasis sind einem Benutzer im allgemeinen nicht bekannt. Die meisten Informationssysteme erlauben einem Benutzer, Teile zu finden und zu manipulieren, wenn dieser deren genauen Namen und Einordnung kennt; sie bieten aber wenig Unterstützung, wenn der Benutzer das betreffende System nicht bereits kennt.

### 6.4 Beispiele von wissensbasierten Systemen mit guter Benutzerschnittstelle

In diesem Abschnitt sollen abrißhaft zwei Prototypen vorgestellt werden, die in unserem Forschungsprojekt INFORM entwickelt wurden.

#### 6.4.1 FINANZ: Wissensbasierte Formularbearbeitung

FINANZ ist ein System für den Sachbearbeiter in einem Büro. Es unterstützt die Bearbeitung von Formularen mit Abhängigkeitsstrukturen (siehe Bild 6.6) und kann als eine wissensbasierte Erweiterung von VISICALC gesehen werden.

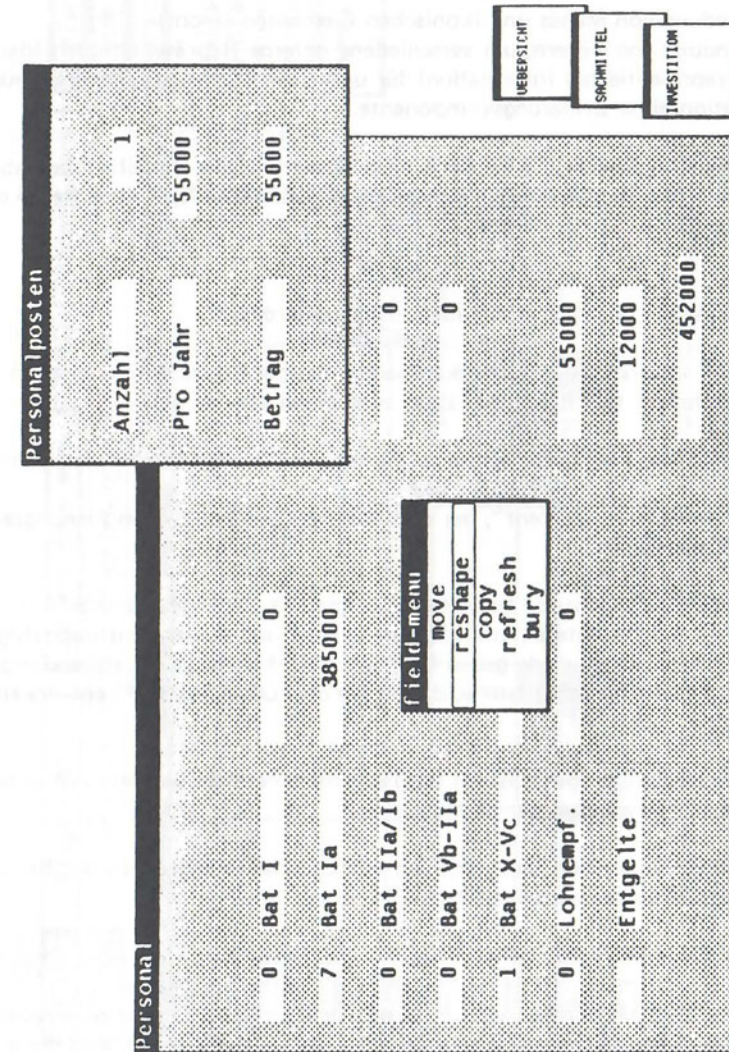


Bild 6.6: FINANZ: Wissensbasierte Formularbearbeitung



Das System befaßt sich mit Finanzierungsplänen für einen Projektantrag und baut auf den folgenden, von uns realisierten Techniken auf:

- Aufbau einer Wissensbasis mit ObjTalk, in der die Abhängigkeitsstrukturen mittels sogenannter „Constraints“ realisiert sind
- Informationsdarstellung in Fenstern; Größe und Inhalt der einzelnen Fenster können durch die direkte Manipulation des Bildschirminhalts verändert werden
- Verwendung von Menüs und ikonischen Elementen (Icons)
- Verwendung von Filtern, um verschiedene externe Repräsentationen (der intern repräsentierten Information) für unterschiedliche Zwecke zu erzeugen
- Integration einer Erklärungskomponente.

Diese Techniken eignen sich für ein breites Spektrum von ähnlichen Aufgabenstellungen (wie z. B. das Ausfüllen eines Lohnsteuerformulars, eines Reisekostenantrags usw.).

#### 6.4.2 PLANER: Computerunterstützte Planungsprozesse

PLANER<sup>10)</sup> unterstützt Informatikstudenten bei der Planung ihres zweiten Studienabschnitts. Das Programm kann verstanden werden als

- ein *Expertensystem*, das bestimmte Funktionen des Studienberaters übernimmt oder
- ein *persönlicher „Assistent“*, der den Benutzer bei komplexen Planungsarbeiten unterstützt.

Die Implementierung wurde auf einem Farbgraphik-Terminal durchgeführt, so daß das Bild 6.7 die Interaktion mit dem System und dessen Leistungsfähigkeit nur unzureichend wiedergeben kann. PLANER benutzt die anwendungs-unabhängige Benutzerschnittstelle, die im DYNAFORM-System<sup>6)</sup> entwickelt worden ist.

Die Wissensbasis (die ebenfalls mit ObjTalk implementiert wurde) wird unter anderem dazu verwendet, dem Benutzer

- Vorschläge zu machen (auf der Basis von Einschränkungen, die der Benutzer dem System mitgeteilt hat);
- ihn bei der Auflösung von Konflikten zu unterstützen;
- multiple Perspektiven mit vom Benutzer definierbaren Filtern zu erzeugen; z. B. sind eine Vielzahl von Informationen mit einer einzigen Vorlesung

| FUNCTION | zeige plan   | select from menu                 | help | undo               | dome              |
|----------|--------------|----------------------------------|------|--------------------|-------------------|
| ACT      | vorschlag => | Pflichtvorlesungen fuer Semester |      |                    |                   |
|          | MO           | DI                               | MI   | DO                 | FR                |
| 7:00     |              |                                  |      |                    |                   |
| 8:00     |              |                                  |      |                    |                   |
| 9:00     |              |                                  |      |                    |                   |
| 10:00    |              |                                  |      |                    | Automaten theorie |
| 11:00    |              | Automaten                        |      |                    |                   |
| 12:00    |              |                                  |      | Automaten          | Dialogsysteme I   |
| 13:00    |              |                                  |      |                    |                   |
| 14:00    |              |                                  |      | Compilerbau        | Formale Semantik  |
| 15:00    |              | Compiler                         |      |                    |                   |
| 16:00    |              | Formale Semantik                 |      | Rechnerarchitektur |                   |
| 17:00    |              |                                  |      |                    |                   |
| 18:00    |              |                                  |      |                    |                   |

Bild 6.7: PLANER: Computerunterstützte Planung



(Pflicht oder Wahl, Dozent, Ort, Zeit, welcher Schwerpunkt usw.) verknüpft, die jeweils nur in einem bestimmten Zusammenhang von Interesse sind;

- lokale (die Gestaltung des Stundenplans für ein bestimmtes Semester) und globale Planungsstrategien (Randbedingungen, die sich durch eine Entscheidung für die Zukunft ergeben) aufeinander beziehen zu können und
- ein persönliches (und damit zielgerichtetes) Informationsangebot zu unterbreiten; dies beruht auf einem Modell des Systems über den Benutzer, das im Laufe der Zeit aufgebaut wird. Dieses Modell kann vom Benutzer jederzeit eingesehen und modifiziert werden und hat bezüglich der dort festgehaltenen Informationen eine ähnliche Funktion wie das Modell, das ein menschlicher Studienberater von einem Studenten aufbaut, der ihn wiederholt aufsucht.

### 6.5 Zukünftige Forschungsschwerpunkte

Obwohl einige Probleme gelöst sind oder besser verstanden werden, Computersysteme „intelligenter“ zu machen, ist damit allenfalls ein Anfang gemacht. Folgende offenen Probleme müssen dringend weiterbearbeitet werden:

1. Die technologische Machbarkeit allein ist nicht ausreichend; es muß mehr Wissen über den Kommunikationspartner Mensch und über informationstechnische Beschreibungen (z. B. von sozio-technischen Strukturen wie einem Büro) erarbeitet werden.
2. Neue Interaktionsmethoden müssen entwickelt werden, die weniger an Hardwareeigenschaften und überholten Programmier Techniken, sondern mehr an natürlichen Kommunikationsweisen des Menschen orientiert sind.
3. Benutzerschnittstellen müssen als gestaltbare Komponenten realisiert werden, die für möglichst viele unterschiedliche Benutzer geeignet sind.
4. Empirische Untersuchungen haben ergeben, daß häufig weniger als die Hälfte der Funktionalität von informationstechnischen Systemen genutzt wird; Verbesserungen in der MCK sind dringend erforderlich, um diesen „Wirkungsgrad“ beträchtlich zu erhöhen.
5. Viele Systemkomponenten (Hilfesysteme<sup>5</sup>), tutorielle Systeme) erfordern, daß ein Problem nicht nur irgendwie gelöst wird, sondern daß die Vorgehensweise des Benutzers mit in Betracht gezogen wird.

6. Benutzerschnittstellen müssen als hochfunktionelle und anwendungsneutrale Einbettungen für möglichst viele Anwendungssysteme konzipiert werden.
7. In Systemen muß vielfältiges Wissen repräsentiert werden. Die Methoden der Wissensakquisition, Wissensrepräsentation und Wissensnutzung müssen weiterentwickelt werden.
8. „Metawissen“ muß repräsentiert werden, das ermöglicht, daß Wissen nicht nur einfach und konsistent hinzugefügt, sondern auch erklärt und neu strukturiert werden kann. Wissensbasierte Systeme sollten auch wissen, was sie nicht wissen.
9. Derzeitige Systeme lösen nur relativ wohlspezifizierte Probleme; weitere Fortschritte (z. B. im Bereich der Büroautomatisierung) erfordern die Modellierung von offenen Systemen; diese wiederum müssen durch den Endbenutzer modifiziert werden können.
10. Systeme sollten in der Lage sein, ihr Handeln und ihre Entscheidungen in größere Zusammenhänge einzubetten.



## Kontakt & Studium

Zum Buch:

Der Leistungsumfang der Hardware-Komponenten bei Rechnern hat bei gleichzeitigem Preisverfall stark zugenommen. Gleichzeitig wächst auch der Verbreitungsgrad dieser Systeme stark an. Damit rückt der Mensch als Benutzer dieser Systeme und als bedeutender Faktor der Mensch-Computer-Interaktion stärker in das Bewusstsein von Anwendern und Herstellern.

Ein Ziel der Forschung und Entwicklung im Bereich der Anwendersysteme ist es, dem Benutzer den Zugang zur Aufgabenbearbeitung und Problemlösung durch verbesserte Benutzerschnittstellen zu erleichtern. Hierin wird im besonderen die Aufgabenstellung der Software-Ergonomie gesehen.

Zur Software-Ergonomie liegen bereits wichtige Untersuchungen und wesentliche Ergebnisse vor. Zur Zeit besteht jedoch noch ein Defizit in der Umsetzung dieser Resultate der Grundlagenforschung in den Bereich der Anwendungen. Ziel dieses Buches ist es daher, sowohl neueste theoretische Entwicklungen anwendernah darzustellen als auch praktische Handlungsanleitungen für die genannte Umsetzung vorzustellen.

Einsatz:  
DV-Führungskräfte, die sich mit Dialoganwendungen befassen; Entwickler von Anwendungssoftware mit Benutzerschnittstellen, Arbeitsgestalter, Organisatoren, DV-Ausbilder und Führungskräfte aus Verwaltung und Datenverarbeitung.

expert  verlag

ISBN 3-8169-0110-7

**Band 200**  
**Kontakt & Studium**  
**EDV**

**Hans-Jörg Bullinger,**  
**Rul Gunzenhäuser**  
**und 6 Mitautoren**

**Software-**  
**Ergonomie**

**Mit 74 Bildern**



**expert verlag.**