



Postvertriebsstück
Gebühr bezahlt

1 Y 10489 F

Herausgeber:
Jo Groebel
Peter Vitouch
Peter Winterhoff-Spurk

MEDIEN PSYCHOLOGIE

Zeitschrift für Individual- und Massenkommunikation

Editorial Board: Albert Bandura · Hubert Feger · Theo Herrmann · Max Kaase ·
Klaus Scherer · Winfried Schulz · Karl-Erik Rosengren · Gery d'Ydewalle

Inhalt: Editorial • *Christian Senn*: Integrierte Aus- und Fortbildung beim Schweizer Fernsehen • *Hermann Körndle / Alf C. Zimmer*: Vom Anfänger zum Könner. Explorierendes Lernen am und mit dem Computer • *Michael Bock / Volker Bussmann / Robin Hörnig*: Wirkungen von Nachrichten und Werbung im Druckmedium und Fernsehen • *Hans-Bernd Brosius / Susanne Kayser*: Der Einfluß von emotionalen Darstellungen im Fernsehen auf Informationsaufnahme und Urteilsbildung • Mitteilungen • Bücher

Sonderdruck

Durch den Buchhandel nicht zu beziehen
© Westdeutscher Verlag GmbH, Opladen 1991

Heft 3/September 1991

Westdeutscher Verlag

Hermann Körndle / Alf C. Zimmer

Vom Anfänger zum Könnler

Eine Studie zum explorierenden Lernen am und mit dem Computer

1. Problemstand

Chapanis (1982) hat bei einer Untersuchung der Bedingungen der Mensch-Computer-Interaktion herausgestellt, daß eine effiziente Kommunikation zwischen Mensch und Computer sowohl im Bedienungs- wie auch im Lernbereich daran scheitert, daß herkömmliche Systeme kaum *Fehlertoleranz* aufweisen. Besonders wenn Rechner zur Unterstützung des Erlernens eingesetzt werden, ist dieser Mangel an Fehlertoleranz aus zwei Gründen fatal: Zum einen verliert der Lernende die Kontrolle über das Lehr-Lernsystem. Zum zweiten kann dieses System in Zustände geraten, die der Lernende nicht mehr diagnostizieren kann und ihm daher erschweren, seine weiteren Aktivitäten sinnvoll zu planen.

Traditionelle Systeme des computerunterstützten Unterrichts haben diesem Problem dadurch abgeholfen, daß sie die Lernsituation so „abgemagert“ haben, daß dem Lernenden nur außerordentlich wenige Verhaltensalternativen in jeder Situation zur Verfügung standen und diese Verhaltensalternativen in keinem Fall zu nicht-diagnostizierbaren Zuständen des Computers führen konnten. Die Hauptnachteile dieser herkömmlichen Systeme, zu dem auch die meisten „computerunterstützten Tutoren“ gehören, bestehen darin, daß die Lernenden kein strukturiertes Wissen über den *Gesamt*problembereich erwerben, sondern üblicherweise nur assoziative Ketten von Einzeldetails. Aus diesem Grunde ist keine Übertragung auf neue Situationen (Transfer) möglich. Schließlich wirken diese Systeme nur in Ausnahmefällen lernmotivierend, da sie üblicherweise den Lernenden entweder unter- oder überfordern – und dies häufig auch noch in raschem Wechsel.

Während herkömmliche Systeme also dem Lernenden ein außerordentlich einschränkendes Korsett aufzwingen, gehen die sogenannten Intelligenten Tutoriensysteme einen diametral anderen Weg hinsichtlich des Fehlerproblems: Sie gehen davon aus, daß jeder Fehler des Lernenden durch das System erstens als Fehler identifiziert werden kann und daß es zum zweiten möglich ist, aufgrund von Fehlern den jeweiligen Wissensstand des Lernenden exakt abzuschätzen. Ist einmal der Wissensstand der Lernenden bekannt, so sollte es möglich sein, ihnen genau die Aufgaben vorzulegen, die gleichermaßen maximal informativ und motivierend sind, d.h. eine Lösungswahrscheinlichkeit von .5 aufweisen. Bisher liegen

allerdings weder überzeugende Algorithmen zur Fehleridentifikation bzw. Diagnose vor, noch erscheint es daher zur Zeit möglich, individuelle Kenntniszustände eindeutig zu modellieren.

Die lern- und motivationspsychologischen Probleme mit dem herkömmlichen Ansatz einerseits und die theoretischen und praktischen Probleme mit dem Ansatz der intelligenten Tutoren andererseits haben dazu angeregt, Situationen aktiven Lernens genauer zu untersuchen. Dabei hat sich herausgestellt, daß „geleitetes Explorieren“ Voraussetzungen für einen effizienten Wissenserwerb und die Übertragbarkeit dieses Wissens auf weitere Problemsituationen schafft. Ein Hauptproblem mit dem Konzept der „geleiteten oder organisierten Exploration“ bestand bisher darin, daß viele Rechner (mindestens zwei pro Lernenden sowie ein großer Zentralrechner) notwendig waren, die außerordentlich schnell miteinander kommunizieren mußten. Durch die Fortentwicklung auf dem Computersektor lassen sich diese Probleme jetzt überwinden.

1.1 Computerunterstützter Unterricht

Auf dem Hintergrund der theoretischen und praktischen Entwicklungen im Bereich der programmierten Unterweisung (siehe hier vor allem die Arbeiten von Skinner, 1968) wurden die Möglichkeiten der elektronischen Datenverarbeitung für eine Verbesserung von Unterrichtstechnologien sehr schnell aufgegriffen. Gegenüber herkömmlichen Unterrichtsprogrammen (linear oder verzweigt) bot die Implementierung von Lernprogrammen auf Computern die folgenden Hauptvorteile:

1. Die Lerngeschwindigkeit konnte individuell angepaßt werden.
2. Eine vollständige Lerngeschichte konnte dokumentiert werden,
3. Aufgrund der jeweiligen Lerngeschichte ließ sich der augenblickliche individuelle Wissensstand eindeutig diagnostizieren.
4. Aufgrund dieser Diagnose ließen sich kompensatorische Maßnahmen planen.

Zu diesen Bereichen liegen zwischenzeitlich umfangreiche theoretische und empirische Arbeiten vor. Im Rahmen dieser herkömmlichen computergestützten Unterrichtstechnologien haben sich besonders die Arbeiten zur Operationalisierung von Lernzielen bzw. zur lernzielorientierten Diagnostik (siehe dazu vor allem Klauer, 1974, 1986) als fruchtbar erwiesen. Einen besonderen Auftrieb haben Programmierungen und Untersuchungen zum Thema computer-aided-learning durch das Projekt „Headstart“ der US-Regierungen unter J.F. Kennedy und L. Johnson bekommen. Dahinter stand die Hoffnung, daß der Einsatz von Computern besonders effektiv für die Behebung von Lerndefiziten sein würde („kompensatorische Erziehung“). Vor allen Dingen aber haben die theoretischen Modellierungen und empirischen Untersuchungen von Atkinson (1970, 1975) gezeigt, daß bei einer Optimierung des Computereinsatzes in Lernsituationen die vorhandenen Unterschiede zwischen den Lernenden noch maximiert werden, da auf jeweils

höheren Lernniveaus die computerunterstützte Lerntechnologie effektiver genutzt werden kann als auf niedrigeren Lernniveaus. Dieses Ergebnis konnte aufgrund theoretischer Annahmen (exponentielle Lernkurven) vorausgesagt und empirisch bestätigt werden. Wenn man diesem Effekt der sich vergrößernden Variabilität in Fertigkeiten entgegenwirken will, kann dies nur dadurch geschehen, daß Lernen auf höheren Niveaus gravierend weniger Lernmöglichkeiten geboten werden. Dies hat aber eine Reduktion der Effektivität des Computereinsatzes in Lernsituationen zur Folge.

Ein weiterer Nachteil des herkömmlichen computerunterstützten Unterrichts (CUU) besteht darin, daß aufgrund der Programm- und Interaktionsstruktur zwischen Computer und Lernenden nicht ausschließlich, aber besonders gut Fakten und einfache Kombinationen von Fakten vermittelt werden können. Abstraktere Fertigkeiten, wie z.B. die induktive Bestimmung von funktionalen Zusammenhängen aufgrund von Daten, elementare mathematische Beweisführungen oder die Generalisierung syntaktischer Regeln werden durch diese Lernprogramme nicht nur *nicht* unterstützt, sondern es zeigt sich sogar eine durchgehende Tendenz zur Schematisierung und Mechanisierung von Faktenverknüpfungen, die einem solchen Lernen von Regeln bzw. abstrakten Strukturen entgegenwirken; dies war aufgrund der Untersuchungen zum 'mental set' (Luchins 1954) zu erwarten gewesen.

Aufgrund der unerwünschten bildungspolitischen Konsequenzen des computerunterstützten Lernens (Erhöhung der Variabilität in Fertigkeiten statt Kompensation) und der negativen Beeinflussung der Fertigungsstruktur durch die computerunterstützte Vermittlung hat sich das klassische Konzept des computerunterstützten Unterrichts (CUU) trotz aller Anfangserfolge bisher nicht als allgemeiner Bestandteil der Unterrichtspraxis durchsetzen können.

Überspitzt kann man sagen, daß ein effektiver Einsatz *nur* von CUU vor allem dann möglich ist, wenn die *Semantik* formaler Sprachen vermittelt wird. Wenn man diese Einschränkung beachtet, läßt sich das Lernverhalten effektiv unterstützen. Eine weitere Einschränkung ergibt sich durch motivationale Faktoren, da im Gegensatz zur Annahme von Skinner dieses Lernverhalten *nicht* durch Selbstverstärkung aufrechterhalten werden kann.

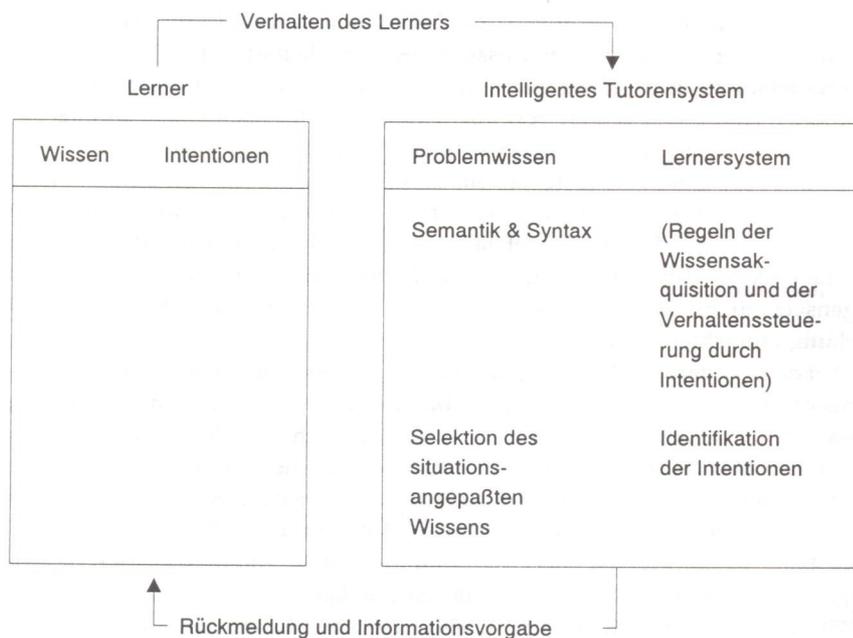
Auf dem Hintergrund der eingeschränkten Anwendbarkeit der herkömmlichen Konzepte des CUU und ihrer negativen bildungspolitischen Konsequenzen ist der Einsatz des CUU seit Anfang der 70er Jahre deutlich zurückgegangen. Erst seit Anfang der 80er Jahre wird die Thematik des CUU erneut aufgegriffen, jetzt aber auf dem Hintergrund neuer theoretischer Konzeptionen zur Struktur des Wissens bzw. zum Prozeß des Lernens: J.S. Brown (1982) und J. Anderson (1981, 1983, 1985, 1986) haben grundlegende Arbeiten zu „Intelligenten Tutoren-Systemen“ vorgelegt, die auf Arbeiten zur künstlichen Intelligenz basieren. Carroll u. Mack (1985) bzw. Zimmer u. Mitarbeiter (1989) haben die Bedingungen für Lernsituationen am Computer entwickelt, die einen *explorativen* Wissenserwerb ermöglichen. Beiden Ansätzen ist die Grundannahme gemeinsam, daß sich Wissen nicht auf Mengen oder Ketten von Fakten reduzieren läßt und daß die eigentliche Über-

prüfung einer erfolgreichen Wissensvermittlung darin bestehen muß, daß *neue* Aufgaben gelöst werden können. Weiter stimmen sie darin überein, daß für den Nutzer die Eingabe/Ausgabe-Einheit funktional der Computer ist, daß also für den Lernenden alle die Eigenschaften des Computers irrelevant sind, die sich nicht in der Gestaltung des Nutzer-Computer-Dialogs auswirken.

1.2 Charakterisierung Intelligenter Tutorensysteme

Im Gegensatz zum herkömmlichen CUU gehen Intelligente Tutorensysteme (eine Übersicht gibt z.B. Woolf, 1989) davon aus, daß komplexe Wissensgebiete nur dann vermittelt werden können, wenn nicht nur Fakten und Faktenketten, sondern die zugrundeliegenden Regelstrukturen Gegenstand der Instruktion sind. Darüber hinaus wird Instruktion dann vor allem wirksam, wenn Gegenstände und Vermittlungsmethode jeweils exakt dem Wissensstand des Lernenden und seinen aktuellen Zielen (Intentionen) angepaßt sind. Damit lassen sich Intelligente Tutorensysteme folgendermaßen schematisch skizzieren (Abb. 1)

Abbildung 1: Grundstruktur von Intelligenten Tutorensystemen



Aus diesem Schema lassen sich die Grundannahmen von Intelligenten Tutorensystemen ableiten:

- 1) Das Verhalten der Lernenden wird nur durch Wissen & Intentionen gesteuert.
- 2) Wissen und Intentionen sind aufgrund des Verhaltens (speziell Informationsnachfrage und Fehler) eindeutig zu identifizieren.
- 3) Die Identifikation basiert auf dem problemorientierten Wissen und dem Lernermodell in Intelligenten Tutorensystemen.
- 4) Nichtübereinstimmungen des beobachteten Verhaltens und des aufgrund des Problems und des Lernermodells vorhergesagten Verhaltens werden als zufällige Störungen interpretiert.

Das zugrundeliegende Modell des Lernens in Intelligenten Tutorensystemen orientiert sich an der Konzeption, daß Verhalten, speziell Wissensakquisition, rational gesteuert ist. Weltwissen wird demnach durch 'Einsicht' (Köhler, 1947) verhaltenswirksam.

Basierend auf der in Abb. 1 gezeigten Grundstruktur sind Intelligente Tutorensysteme u.a. für die Bereiche 'Einfache Beweisführungen in euklidischer Geometrie', 'Programmieren in der Programmiersprache LISP'; 'Erlernen algebraischer Regeln' und für technische Spezialanwendungen entwickelt worden. Wenn überhaupt ihre Effektivität überprüft worden ist, dann durch den Vergleich mit individualisiertem Unterricht, wobei allerdings weder die Einflußgrößen „Umfang“ noch „Struktur“ des vermittelten Themas überprüft worden sind. Zudem sind die dem Lehrverhalten zugrundeliegenden didaktischen Konzeptionen der Lehrenden nicht definiert worden. Insgesamt deuten zwar die Ergebnisse darauf hin, daß Intelligente Tutorensysteme effizienter sind als herkömmliche CUU, aber weniger als persönlicher individueller Unterricht.

Eine grundsätzliche Kritik am Ansatz der Intelligenten Tutorensysteme besteht darin, daß sie jeweils die Existenz zweier vollständiger Expertensysteme voraussetzen – eines für das Problemgebiet und ein zweites für das Lernverhalten. Eigentlich macht aber ein vollständiges Expertensystem für das jeweilige Problem die Instruktion von Lernenden über dieses Problem überflüssig: Warum Bedienungspersonal oder Programmierer ausbilden, wenn diese maximal so gut sein können wie ein Rechner auf dem die jeweiligen Expertensysteme installiert sind? Dieser Einwand könnte nur dadurch behoben werden, daß man die *paradigmatische* Wirksamkeit von Intelligenten Tutorensystemen nachweist, die den Transfer auf andere Bereiche ermöglicht.

Noch gravierender sind die Einwände gegen die *impliziten Grundannahmen* Intelligenter Tutorensysteme:

- 1) Es liegen zwar üblicherweise Rahmenintentionen beim Lernen vor, diese sind jedoch nur in den seltensten Fällen so expliziert und differenziert, daß sie auch spezifisches Frage- und Antwortverhalten determinieren. Damit dürften die Intentionen der Lernenden nur schwer zu identifizieren sein.

- 2) Wissensdomänen sind speziell während des Akquisitionsvorganges nur unvollständig geordnet, daher ist aus dem spezifischen Lernerverhalten kein hinreichend genauer Schluß auf seinen Kompetenzstand möglich.
- 3) Aufgrund von Verhaltensdaten ist eine Trennung von Fehlerkategorien (Fehlkonzeptionen, Versuch-und-Irrtum, Flüchtigkeit etc.) nur partiell möglich, daher lassen sich auf dieser Basis keine individuellen Lernermodellierungen durchführen.
- 4) Es muß angenommen werden, daß die internen Zustände des Lernenden *nur* durch die Interaktion mit dem Intelligenten Tutorsystem verändert werden.

1.3 Lernen durch gesteuerte (organisierte) Exploration

Aufgrund der technischen Entwicklungen, die einerseits zur Herstellung preiswerter Computersysteme geführt haben und andererseits neue Kommunikationssysteme anbieten, ist es seit einiger Zeit möglich, explorierendes Lernen auf dem Computer zu unterstützen. Dabei geht man zunächst von dem Konzept aus, daß es für solche Lernsituationen optimal wäre, Wissen in diesem System so zu speichern wie Waren in einem Warenhaus, wo die Käufer durch Stöbern bzw. gezieltes Fragen die Waren finden, die sie suchen. Analog kann das Lernverhalten in computerunterstützten Systemen modelliert werden. Es stellt sich dann heraus, daß ein derartiger Wissenserwerb nur effizient wird, wenn der Lernende eine Rahmenkonzeption für den Bereich mitbringt, in dem er oder sie detailliertes Wissen erwerben will.

Eine erste Strukturierung dieses explorierenden Lernens am Computer wurde von J. S. Brown (1975) mit dem Konzept der „increasingly complex micro worlds“ (ICM) vorgeschlagen: In diesem Konzept wird davon ausgegangen, daß sich die Lernenden in einer einfachen Situation, für die sie ein Rahmenkonzept haben, nicht nur das notwendige Detailwissen durch Exploration erwerben, sondern darüberhinaus auch die Rahmenkonzeption für eine geringfügig komplexere Lernsituation erwerben. Haben sie das Wissen der einfachen Situation erworben, wird zur nächsten leicht komplexeren übergegangen u.s.w. Dieses Konzept ist prinzipiell nur dann anwendbar, wenn der zu vermittelnde Stoff streng hierarchisch gegliedert ist. Weicht die Strukturierung des Lernmaterials jedoch von der strengen Hierarchie ab, dann lassen sich keine ICMs mehr konstruieren. In seiner Analyse der ICMs weist jedoch J. S. Brown auf einen Aspekt hin, der eine Verallgemeinerung dieses Konzepts ermöglicht: Die Bedeutung des „learning by doing“, d. h. die Lernenden erschließen sich die strukturellen Regeln des Lernstoffs dadurch, daß sie ihn systematisch manipulieren. Auf dieser Grundlage sind sie in der Lage, systematische Hypothesen zu entwickeln und diese auf neue Situationen anzuwenden. Die Modellierung einer solchen Lernsituation kann analog der Theorie der Schemaintegration (Zimmer, 1986) erfolgen.

Ein Schema kann durch folgende Bestandteile definiert werden:

- 1) eine Menge von basalen Einheiten („primitives“), die im gegebenen Kontext nicht weiter analysierbar sind;
- 2) eine Menge von Organisationsregeln;
- 3) eine Menge von zulässigen Transformationen, die die Klasse von Invarianten der betrachteten Objekte erzeugt.

Beispiel:

Ein Gesicht besteht aus einer Menge basaler Einheiten, wie Augen, Nase, Mund, Ohren, usw. Die Organisationsregeln beinhalten die geometrische Anordnung dieser Einheiten, so daß daraus ein Gesicht entsteht. Eine mögliche zulässige Transformation könnte beispielsweise die Zahl der Augen betreffen: Auch „Gesichter“ mit drei oder mit einem Auge (Zyklopengesicht) würden zu der Klasse der Gesichter zählen.

Eine wichtige Konsequenz dieser Definition ist, daß schematisch angeordnetes Wissen nicht allein aus einer Menge basaler Einheiten und den zugehörigen Organisationsregeln für ihre Anordnung besteht, sondern daß dazu auch die Menge der zulässigen Transformationen dieses Wissens beachtet werden muß.

Explorierendes Lernen kann somit als systematischer, aber nicht notwendigerweise vollständiger Erwerb eines Schemas verstanden werden. Die beim Lernen miterworbenen zulässigen Transformationen bestimmen damit mögliche Transfersituationen für das so erlernte Wissen. Welche Schemata beim explorierenden Lernen in welcher Reihenfolge erarbeitet werden, bestimmt ausschließlich der Lernende. Explorierendes Lernen bedeutet also nicht wahl- und zielloses Aufpicken von Schema-Details.

2. Ein Tutorsystem für das explorierende Lernen

Während inzwischen eine Reihe von Tutorsystemen existieren, die die grundlegenden Gedanken intelligenter Tutorsysteme mehr oder weniger realisieren, z.B. der Geometrie-Tutor von Anderson et al. (1985), STEAMER von Hollan et al. (1984) oder SOPHIE von Brown et al. (1982), existiert für den Bereich des explorierenden Lernens ein solches System nicht. Für seine Realisierung sind folgende Konstruktionsanforderungen zu berücksichtigen:

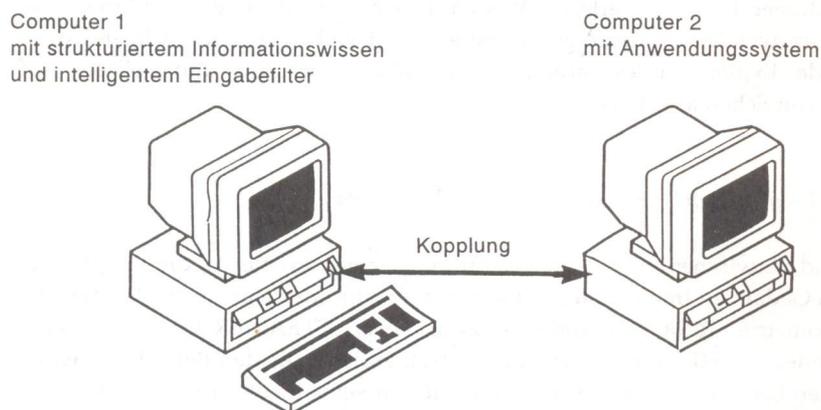
- 1) Strukturierte Stoffanordnung und Stoffübersicht mit der Möglichkeit, diesen Stoff beliebig zu bearbeiten: Während die Strukturierung des Stoffs den Erwerb und die Übertragbarkeit von Rahmenkonzeptionen auf neue Lernsituationen unterstützt, ermöglicht der beliebige Zugriff auf den Lernstoff dem Lernenden das Explorieren.
- 2) Unterschiedliche Grade von Unterstützung der Lernenden durch das System: Die Lernenden können sich den Grad der Unterstützung durch das Lernsystem auswählen. Im einen Extremfall erhält der Lernende keine Unterstützung, im anderen Fall der totalen Unterstützung bleiben dem Lernenden keine Eingriffsmöglichkeiten mehr.

- 3) **Kopplung von Wissenserwerb und Wissensanwendung:** Die Lernenden haben jederzeit die Möglichkeit, Wissen, das ihnen das Tutorsystem liefert, in der Originalsoftware anzuwenden. Dazu steht das Tutorsystem den Lernenden auch wissensspezifische Beispiele und Aufgaben zur Verfügung.

Diese Anforderungen wurden bei einem Tutorsystem realisiert, das dem Erlernen der Datenbanksoftware dBase dient. Dieser Aufgaben- und Anwendungsbereich wurde gewählt, da er sowohl weitverbreitet wie auch in seiner zugrundeliegenden Struktur einfach zu verstehen ist. Man kann sogar davon ausgehen, daß jeder, der einmal mit Listen gelernt hat, für dBase die kognitiven Rahmenkonzeptionen mitbringt. (Allerdings ist darauf hinzuweisen, daß sich die Module dieses 2-Prozessor-Tutorsystems auch für andere Anwendungsbereiche eignen und nicht ausschließlich an die Software dBase gebunden sind.)

Das auffälligste Kennzeichen dieses Tutorsystems ist die Verwendungen zweier gekoppelter Rechner.

Abbildung 2: 2-Prozessor-Tutorsystem, bei dem 2 Rechner über die serielle Schnittstelle miteinander gekoppelt sind.



Auf dem rechten Rechner ist die zu erlernende Anwendersoftware installiert, auf dem linken Rechner die Tutorsoftware. Sie erlaubt dem Lernenden strukturiertes Informationswissen explorativ abzufragen und in vier verschiedenen Lernarten in der Originalsoftware anzuwenden und zu erproben. Das Tutorsystem kontrolliert dabei die Tastatureingaben auf Fehler, gibt sie bei Fehlerfreiheit an die Anwendersoftware weiter oder gibt Rückmeldung über den Fehler. An dieser Stelle kann der Lernende durch weitere Bearbeitung von Anleitungstexten oder Beispielen das zur Fehlerbehebung notwendige Wissen erwerben, ohne negative Konsequenzen, wie z.B. einen Systemabsturz oder Datenverlust, erwarten zu müssen.

Im Detail wurden bei der Systemgestaltung folgende Punkte beachtet:

- a) Explorationsmöglichkeiten
Das Tutorsystem gibt keinen Lernweg vor. Ein einmal gewählter Lernweg kann jederzeit abgebrochen werden. Die vollständige Bearbeitung eines Kapitels wird angezeigt.
- b) Fehlertoleranz
Die Eingaben des Lernenden werden nur dann an den dBase-Rechner weitergegeben, wenn sie entweder fehlerfrei sind oder nur solche Fehler beinhalten, die der Lernende in dBase korrigieren kann, wie z.B. Tippfehler beim Ausfüllen von Feldern. Eine fehlerhafte Eingabe, die dBase in einen nicht mehr zu korrigierenden Zustand versetzen würde, werden vom Tutorrechner abgefangen und nicht an den dBase-Rechner weitergeleitet.
- c) Verschiedene Lernarten
Das Tutorsystem bietet dem Lernenden vier verschiedene Lernarten an, sein Wissen in der Originalsoftware zu erproben. Die vier Lernarten sind die Vorführung, das schrittweise Bearbeiten eines Beispiels, die Durcharbeitung einer Übung und die Abarbeitung einer Aufgabe. Diese Lernarten unterscheiden sich bezüglich der Explorationsmöglichkeiten des Lernenden und durch die Unterstützung, die der Lernende erhält.

Lernart	Unterstützung durch den Tutor	Explorationsmöglichkeit
<ul style="list-style-type: none"> - Vorführung - Beispiel - Übung - Aufgabe 	<p>vollständig</p>  <p>keine, nur dBase-Voreinstellung bei Neubeginn einer Bearbeitung</p>	<p>keine</p>  <p>dBase vollständig zugänglich</p>

In der Lernart „Vorführung“ steht dem Lernenden keine Explorationsmöglichkeit zur Verfügung, er kann nur den Ablauf der Bearbeitung eines Datenbankkonzepts auf der Originalsoftware verfolgen. In dieser Lernart unterstützt das Tutorsystem den Lernenden vollständig. Der entgegengesetzte Fall liegt für die Lernart „Aufgabe“ vor: Hier kann der Lernende die Originalsoftware vollständig explorieren, allerdings erhält er vom Tutorsystem dafür keinerlei Unterstützung. Auch Fehler werden nicht mehr abgefangen.

Folgende Abbildungen zeigen charakteristische Bildschirminhalte für die verschiedenen Lernarten zum Wissensbereich „Datei anlegen“. Dabei zeigt der linke Teil der Abbildungen den Bildschirminhalt des Tutors, der rechte Teil die zur jeweiligen Lernsituation gehörige dBase-Oberfläche.

Die Tutorinformation wird in 3 Zonen am Bildschirm dargestellt. Am linken Rand des Bildschirms befinden sich die Menüs, über die eine Lerneinheit und die

MAPITEL
 Tutor
 Datenbank
 Assistent
NEUE DATEI
 Zusammenfassung
 Ende

Beispiel Datei anlegen - INFORMATION
 Legen Sie nun eine neue Datenbank PERSONAL an.

NAME	
VORNAME	28
ZEICHEN	15
CEREBITUM	
LEDIC	Logisch
TELEFON	Zeichen 18
GEHALT	numerisch 7, Dez. 2
BEARTEIL	Memo

Stellen Sie mit der Taste <BASE> Verbindung zu DBASE her.
 Mit <Zurück> kommen Sie zurück zum Menü LERNWÄRT.

Beispiel Datei anlegen - BEARBEITUNG
 Geben Sie den Dateinamen PERSONAL ein und bestätigen Sie die Eingabe mit <OK>.

NEUE DATEI
 Eingeben
 Ausgeben
 Wiederholung
 =>MAPITEL

Auswahl **DB** Modus Position Extrakt Organisation Ändern Dienste **BEFEHLE**

Datenbank
 Bericht
 VIEW
 QUERY
 Report
 Label

Geben Sie den Dateinamen ein: **PERSONAL**

Befehl: **CREATE A:**
ASSIST **<OK>** **<DATE 1/5**
 Eingabe einer neuen Wert. Bestätigen mit <OK>.
 Geben Sie einen Dateinamen ein.

Abbildung 4: Lernart Beispiel: Der Lernende kann „Schritt für Schritt“ eine neue Datei anlegen.

KAGITEL
Tutor
Datenbank
Assistent
Neue Datei
Zurückfassung
Ende

NEUE DATEI
Anlegen
Eingeben
Ausgeben
Wiederholung
=>KAGITEL

LEHRT
Vorführung
Beispiel
Übung
Aufgabe
=>NEUE DATEI

Übung Datei anlegen - INFORMATION
Legen Sie nun eine neue Datenbank BUCH mit folgender Struktur an:

TITEL
VERFASSER
PREIS
KAUFDATUM
INHALT

HILFE
Bestätigen Sie Laufwerk A mit (Get).
Mit (Zurück) verlassen Sie die Hilfe.

her.

Stellen Sie
Mit (Z) (Notiz) (Zurück)

Übung Datei anlegen - BEARBEITUNG
Sie sind jetzt mit dBASE verbunden.
Mit (Hilfe) bekommen Sie an Tutor zusätzliche Hilfe angezeigt.
Mit (Zurück) kommen Sie zurück zur INFORMATION.
Wenn Sie einen Fehler machen, beantworten Sie bitte die vom Tutor gestellten Fragen.

Auswahl **Modus** Position Extrakt Organisation ändern Dienste **ZURÜCK**

Datenbank
Format
VIEW
QUERY
Report
Label

A:
B:
C:
D:
E:

Befehl: CREATE
ASSIST **(Get)** Markierungsballen - H. Auswahl - **(Z)**. Menüauswahl - **(Z)**.
Wählen Sie ein Laufwerk.

Abbildung 5: Lernart Übung: Der Lernende kann unter der Fehlerkontrolle des Tutors die Übung bearbeiten: Fehler werden rückgemeldet und verlangen vom Lernenden eine Angabe über die vermutete Fehlerursache. Aufruf kontextspezifischer Hilfe ist möglich.

KAPITEL
Tutor
Datenbank
Assistent
NEUE DATEN
Zusammenfassung
E n d e

Restliche Bytes: 3968

CURSOR: (← →)		Einfügen		Löschen		Feld auf	
Zeich.:	Home End	Zeich.:	Ins	Zeich.:	Del	Feld ab	↓
Spalte:	↵ ↶ ↷	HILFE:	F1	Wort:	↵	End/Sichern:	End
			F1	Feld:	↵	Abbruch:	Esc

Feldname	Typ	Länge	Dez
1 NAME	Zeichen	28	
2 UORNAME	Zeichen	28	
3 TELNR	Zeichen		

NEUE DATEN
EINGEBEN
Ausgeben
Wiederholung
=>KAPITEL

LEHRE
Vorführung
Beispiel
Übung
NEUE DATEN
=>KAPITEL

Legen Sie die Datenbank TELEFOR an, mit deren Hilfe Sie ein Telefonverzeichnis von Ihren Bekannten anlegen, ohne die einzelnen Mäusen und Daten einzugeben.

Stellen Sie mit der Taste <BASE> Verbindung zu <BASE> her. Mit <Zurück> kommen Sie zurück zum Menü LEHRE.

Aufgabe <BASE> Menü - BEARBEITUNG

Sie sind jetzt mit <BASE> verbunden. Der Tutor bietet Ihnen hier keine Unterstützung. Sie haben aber die Möglichkeit, in Assistenten Hilfe von <BASE> mit der Taste <F1> anzufordern. Wenn Sie sich nicht mehr zurecht finden oder aufhören wollen, drücken Sie die Taste <Zurück>. Sie sind dann wieder mit dem Tutor verbunden, der Sie zu Ihrer Vorgehensweise befragt.

[Text] <otiz> <Zurück>

Feldname	Typ	Länge	Dez
1 NAME	Zeichen	28	
2 UORNAME	Zeichen	28	
3 TELNR	Zeichen		

LEHRE **BASE** **TELEFOR** **BASE** **BASE**

Drücken Sie die Leertaste, um den Feldtyp zu ändern.
Zeichenfelder enthalten Text-Informationen einer festgelegten Länge.

Abbildung 6: Lernart Aufgabe: Vollständige Verfügbarkeit der Originalsoftware für den Lernenden ohne Kontrolle durch den Tutor.

Lernart angewählt werden kann. Die Fläche rechts davon unterteilt sich in einen Informationsbereich mit Erläuterungen und Anleitungen sowie ein Fenster, das die für die direkten Schritte mit dBase notwendige Information enthält. Die Informationsdarstellung erfolgt mit Fenstern, die die Texte beinhalten. Eine Kopfzeile enthält den Titel des Fensters, eine Fußzeile gibt an, welche Funktionstasten genutzt werden können und ob Texteingabe möglich ist.

Wie die Abbildungen zeigen, sind die Informationen, die sich die Lernenden im Tutorsystem geben lassen können, direkt mit einem konkreten Bearbeitungsablauf am zweiten Rechner umsetzbar. Sollten die Lernenden dennoch Hilfe benötigen, steht sie ihnen kontextspezifisch zur Verfügung (vgl. Abbildung 5). Darüberhinaus besteht die Möglichkeit, sich im jeweiligen Kontext Notizen z.B. über bestimmte Lösungswege zu machen.

3. Eine experimentelle Studie zum explorierenden Lernen

Eine experimentelle Untersuchung explorierenden Lernens setzt eine Lernumgebung voraus, deren Eigenschaften oben aufgelistet wurden. Da bisher keine experimentellen Daten über Art und Verlauf explorierenden Lernens vorliegen, ist die Erhebung solcher Daten in einer entsprechenden Lernumgebung angezeigt. Wichtig ist dabei auch die Erhebung von Vergleichsdaten mit einem Tutorsystem, das explorierendes Lernen nicht zulässt. Im vorliegenden Fall bietet sich als Vergleichstutor der für dBase III Plus erhältliche Tutor an, der in herkömmlicher Weise nur den aneinandergereihten Erwerb von Fakten zulässt. Beide Tutoren umfassen denselben zu erlernenden Befehlsumfang.

Folgende Tabelle listet die Eigenschaften der in der experimentellen Studie benutzten Tutorsysteme auf:

	2-Prozessor-Tutorsystem	dBase-Tutor vom Hersteller Ashton-Tate
Benutzeroberfläche	Tutor in Fenstertechnik Anwendung im dBase	Ausschnitte aus der Originalsoftware
Funktionsumfang der zu erlernenden Software	voller Funktionsumfang in dBase zugänglich	stark eingeschränkt, oft nur einzelne Tastendrucke möglich
Lernwege	explorierendes Lernen möglich	kapitelweises, lineares Vorgehen
Fehlerbehandlung	lernartabhängig	keine, da Eingabe festgelegt
Hilfe	kontextspezifisch	keine
Reflexion des gelernten Stoffes	- nach der Aufgabebearbeitung - nach dem Auftreten von Fehlern - Notizmöglichkeit	Wiederholung am Kapitelende möglich

Im Rahmen einer experimentellen Studie zum explorierenden Lernen versuchten wir folgende Fragenkomplexe zu klären:

- 1) Wie agieren Lernende in einer Lernsituation, die explorierendes Lernen erlaubt?
- 2) Welchen Effekt hat explorierendes Lernen für den Transfer auf neue Situationen und Aufgaben?

Um für die Lernenden gleiche Ausgangsbedingungen für die Arbeit mit dem jeweiligen Tutorsystem zu schaffen, wurde die Lernsituation mit dBase in ein Schulungskonzept eingegliedert. Dabei wurden vor dem eigentlichen Lernexperiment elementare Hard- und Softwarebegriffe mit Schulungsunterlagen erläutert, der Gebrauch der Tastatur an einer Textverbesserungsaufgabe geübt und Fragen zum bisherigen Umgang der Lernenden mit elektronischer Datenverarbeitung bearbeitet.

Die Probanden waren männliche Auszubildende der metallverarbeitenden Industrie. Sie hatten alle keine Vorerfahrung in der Nutzung der Datenbank-Software dBase. Zwölf der Probanden arbeiteten mit dem 2-Prozessor-Tutorsystem, acht mit dem zu dBase gehörenden Tutor. Die Teilnahme am Experiment erfolgte freiwillig während der Arbeitszeit.

Beide Lerngruppen hatten nach der Arbeit mit dem jeweiligen Tutorsystem acht Aufgaben in dBase zu bearbeiten. Sie waren aufgrund von Erfahrungen in dBase-Kursen so ausgewählt, daß über die Aufgaben eine mittlere Lösungswahrscheinlichkeit zu erwarten war. Fünf der Aufgaben waren ausschließlich mit dem durch den Tutor vermittelten Stoff zu lösen, während die drei anderen Aufgaben nur durch Transfer gelöst werden konnten.

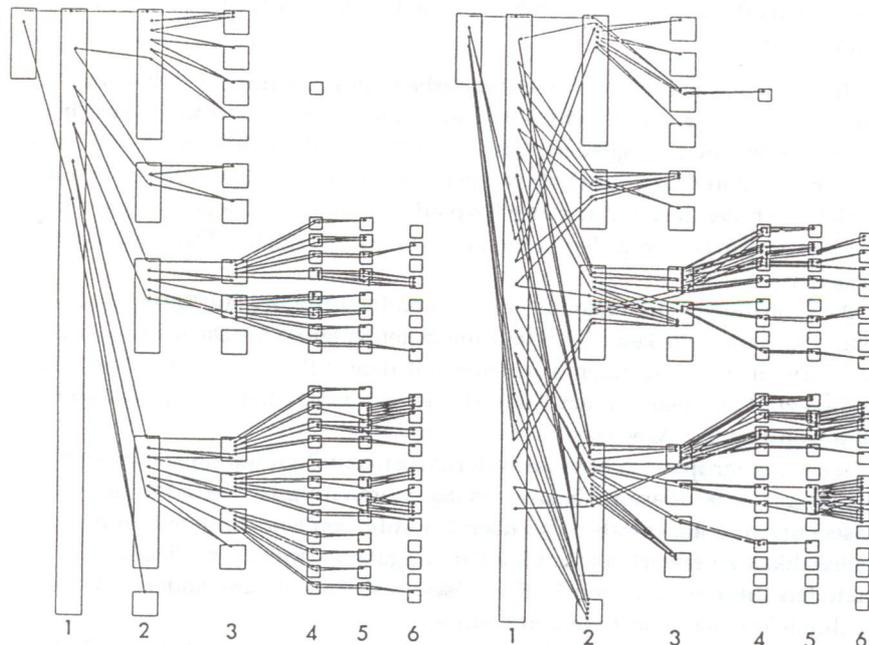
Die Gruppe, die mit dem 2-Prozessor-Tutorsystem lernte, löste im Mittel 6 Aufgaben (Standardabweichung 1,3) während die Gruppe mit dem Standard-Tutor im Mittel nur 1,6 Aufgaben (Standardabweichung 1,8) löste. Für die Transferaufgaben findet man vergleichbare Mittelwertsunterschiede von 1,8 und 0,3 gelösten Transferaufgaben (Standardabweichung 0,45 vs. 0,3).

Bei der Anzahl der gelösten Aufgaben zeigt sich klar, daß durch die Lernumgebung des 2-Prozessor-Systems eine Effektivierung des Stofferwerbs möglich ist. Vor allem der Unterschied bei den gelösten Transferaufgaben macht deutlich, daß diese Effektivierung nicht ausschließlich als Mengeneffekt zu verstehen ist, sondern daß beim Lernen mit dem 2-Prozessorsystem auch allgemeine bzw. generalisierbare Konzepte erworben werden, die der Anwendungssoftware zugrundeliegen.

Um genauer zu klären, welches Lernverhalten Probanden am 2-Prozessor-Tutorsystem zeigen, wurden detailliert die Lernwege der Probanden untersucht. Die Abbildungen 7a und 7b zeigen beispielhaft die Lernwege zweier Probanden, die systematisch bzw. unsystematisch den Lernstoff bearbeitet haben.

Allgemein weisen die Lernverläufe eine hohe Varianz im Explorationsverhalten auf. Im einzelnen bedeutet das, daß die Lernenden die Möglichkeit, in der Lernstoffmenge beliebig zu wählen, extensiv bei der Lern-Kapitelauswahl nutzen, aber

Abbildung 7a, b: Zwei Extrembeispiele für explorierendes Lernen a) systematisch, b) unsystematisch. Die Ziffern geben die verschiedenen Ebenen des hierarchisch angeordneten Lernstoffes an:



- 1 = Ausgangsmenu (entsprechend Fenster links oben in Abb. 3-6)
- 2 = Informationskapitel über Datenbanksysteme (entsprechend Fenster Mitte links)
- 3 = Detailinformationen über die Kapitel in Ebene 2 (großes Informationsfenster)
- 4 = Lernarten: Vorführung, Beispiel, Übung, Aufgabe (entsprechend Fenster links unten)
- 5 = Anwendung des Stoffes in dBase (rechter Bildschirm)
- 6 = Fehler in dBase

nach erfolgter Kapitelwahl die Unterkapitel und ihre Anwendung in der dBase-Software relativ systematisch nutzen.

Der Lernerfolg der Gruppe, die mit dem 2-Prozessor-System arbeitete, kann also darauf zurückgeführt werden, daß sie beim Lernen nicht nur isolierte Konzepte, sondern die zur Lösung von Datenbankaufgaben notwendigen Konzeptabfolgen erworben haben. Für diese Lerngruppe spielt es dann bezüglich späterer Aufgaben oder Anwendungen keine Rolle, ob sie den Stoff systematisch oder unsystematisch bearbeitet haben.

So gesehen kann exploratorisches Lernen als Erwerb der Invarianten eines Konzepts (oder auch Schemas) verstanden werden, indem einerseits die Anwend-

barkeit eines Konzepts auf verschiedene Aufgaben vom Lernenden getestet, andererseits irrtümlich angenommene Konzepteigenschaften verworfen werden.

4. Diskussion

Die Wirksamkeit eines Lernsystems, das explorierendes Lernen erlaubt, läßt sich auf folgende Einflußgrößen zurückführen:

- a) Exploration
 - Durch Explorieren finden die Lernenden die Invarianten im Lernstoff, indem sie intentionale Fehler begehen und aus diesen Konsequenzen lernen.
 - Die im Tutor dargebotene Information kann direkt in der Originalsoftware angewandt werden.
 - Diese Anwendung der Tutorinformation geschieht in Abhängigkeit vom Lernstand der Lernenden.
- b) Transfer
 - Die Lernenden erwerben nicht nur starre Kommandofolgen
 - Die Trennung von Tutoroberfläche und Originalsoftware-Oberfläche ermöglicht den Lernenden die Trennung von Lern- und Anwendungssituation.
- c) Gedächtnishilfe
 - Das System motiviert die Lernenden, eigene Systembeschreibungen (mögliche andere Anwendungen, Analogien, Formulierungen) im System zu speichern und für spätere Anwendungsfälle bereitzuhalten.
 - Das System erlaubt Lösungswege zu sichern, die nicht mehr rational rekonstruiert, sondern nur noch erinnert werden können.

Die Lernwirksamkeit des 2-Prozessor-Tutorsystems bestätigt die theoretischen Überlegungen, daß aktive Lernsituationen effizienten Wissenserwerb und die Übertragbarkeit dieses Wissens auf neue Situationen erlauben. Unter pragmatischen Gesichtspunkten fordert sie die Überlegung heraus, ob in allen Anwendungsfällen, für die intelligenten Tutorsysteme entwickelt werden, der Aufwand im Vergleich zur beschriebenen Alternative gerechtfertigt ist. Denn eines sollte man bei aller technologischen Entwicklung nicht vergessen: Lernen bedeutet für die Lernenden selbst aktiv zu sein, der Nürnberger Trichter (Carroll, 1990) ist noch nicht erfunden.

Literatur

- Anderson, J.R. (1981). *Cognitive Skills and their Acquisition*. Hillsdale, N.J.: Lawrence Erlbaum.
- Anderson, J.R. (1983). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.

- Anderson, J.R., Boyle, C. & Yost, G. (1985). *The Geometry Tutor*. Proceedings of the International Joint Conference on Artificial Intelligence. Los Angeles, CA.
- Anderson, J.R., Reiser, B. (1986). The LISP Tutor. *Byte*, 10, pp. 159-175.
- Atkinson, J.W. (1975). *Einführung in die Motivationsforschung*. Stuttgart: Klett.
- Atkinson, J.W., Birch, D. (1970). *The Dynamics of Action*. New York: Wiley.
- Brown, J.S., Burton, R.R. (1975). Multiple representation of knowledge for tutorial reasoning. In D. Bobrow & A. Collins (eds.), *Representing and Understanding: Studies in Cognitive Science*. New York: Academic Press.
- Brown, J.S., Bell, A. (1982). SOPHIE: A Sophisticated Instructional Environment for Teaching Electronic Troubleshooting. In D. Sleeman and J.S. Brown (eds.), *Intelligent Tutoring Systems*. Cambridge, MA: Academic Press.
- Carroll, J.M. (1990). *The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill*. Cambridge, MA: The MIT Press.
- Carroll, J.M., Mack, R.L., Lewis, C.H., Grischowsky, N.L. & Robertson, S.R. (1985). Exploring exploring a Word processor. *Human-Computer Interaction*, 1, 283-307.
- Chapanis, A. (1982). Computer and the Common Man. In A. Kaschau, R. Lachman and K.R. Langhery (eds.), *Information Technology and Psychology: Prospects for the Future*. Houston Symposium 3. Houston: Praeger Publisher, pp. 106-132.
- Hollan, J., Hutchins, E. & Weitzman, L. (1984). STEAMER: An Interactive Inspectable Simulation-Based Training System. *AI Magazine*. Summer.
- Klauer, K.J. (1974). *Methodik der Lehrzieldefinition und Lehrstoffanalyse*. Düsseldorf: Schwan.
- Klauer, K.J. (1986). *Kriteriumsorientierte Tests: Lehrbuch der Theorie und Praxis lehrzielorientierten Messens*. Göttingen: Hogrefe.
- Köhler, W. (1947). *Gestalt Psychology: An introduction to new concepts in modern psychology*. New York: Liveright Publ. Corp.
- Körndle, H., Ziegler, S. & Zimmer, A.C. (1988). *Lernen durch Explorieren am Computer ein 2-Prozessor-System für dBase*. Referat auf der 30. Tagung experimentell arbeitender Psychologen in Marburg.
- Luchins, A. S. (1954). *A variational approach to the vole of set in problem solving*. Proceedings of the 14th International Congress of Psychology.
- Skinner, B.F. (1968). *The Technology of Teaching*. New York: Meredith Corporation.
- Woolf, B. (1989). Intelligent Tutoring Systems: A Survey. In H.E. Shrobe (ed.), *Exploring Artificial Intelligence*.
- Zimmer, A.C. (1986). The economy principle, perceptual mechanisms and automated cognitive processes. *Gestalt Theory*, 8, 174-185.
- Zimmer, A.C., Körndle, H. (1989). Designing Dialogue Processes in a Dynamic Human-Computer Interaction. In F.Klix, N.A. Streitz, Y. Waern & H. Wandke (eds.), *Man-Computer Interaction Research. MACINTER II*. (S. 501-509). Amsterdam: North-Holland.

Anschrift der Verfasser: Dr. Hermann Körndle/Prof. Dr. Alf C. Zimmer, Institut für Psychologie, Universität Regensburg, Universitätsstraße 31, 8400 Regensburg