

DIE PEIRCE-SEMIOTIK ALS ANSATZPUNKT FÜR EINE DIDAKTIK DER INFORMATIONSVERARBEITUNG

Semiotische Fragestellungen zur Mensch-Maschine-Kommunikation

Lange Zeit hat die semiotische Forschung einen Problembereich vernachlässigt, der mittlerweile die berufliche Lebenswelt der meisten Menschen betrifft: die Mensch-Maschine-Kommunikation, und insbesondere die Notwendigkeit, informationstechnische Systeme verstehen und anwenden zu können. Seit einigen Jahren steigt jedoch das Interesse an semiotischen Modellen im Bereich der Mensch-Maschine-Kommunikation (MMK). Es erwächst hauptsächlich aus den Disziplinen Informatik, Kognitionswissenschaft und Linguistik, die darüber hinaus die tragenden Säulen der Künstlichen Intelligenz-Forschung sind. Hier lassen sich zwei Schwerpunkte erkennen:

- Die von Peirce formulierte Inferenzart "Abduktion" spielt eine zunehmend große Rolle im Bereich "automatic reasoning".¹
- Semiotische Ansätze fließen ein in Arbeiten zur Repräsentation von (sprachlichem) Wissen bei Mensch und Maschine.² Im Mittelpunkt steht dabei häufig der Bedeutungsbegriff, zum Beispiel für die Probleme maschineller semantischer Sprachanalyse.³

Doch nicht nur Protagonisten der Künstlichen Intelligenz-Forschung greifen zeichentheoretische Fragestellungen auf; auch ihre prominenten Kritiker wie Dreyfus (1985) nutzen - besonders die phänomenologisch gestützten - Erkenntnisse der Semiotik, vor allem, um menschliche Erkenntnis- und Repräsentationsformen gegen maschinelle Intelligenz abzugrenzen.

Den zitierten Arbeiten sind zwei Aspekte gemeinsam: a) Sie berufen sich fast alle auf Charles Sanders Peirce, was dessen herausragende Bedeutung für die Semiotik und das Themengebiet "MMK" unterstreicht.⁴ b) In ihnen ist das Forschungsinteresse erkennbar, die Leistungsfähigkeit von Computern zu verbessern - in Anlehnung an oder Abgrenzung von menschlichen Kognitionsvermögen. Man schaut auf die Mensch-Maschine-Kommunikation aus dem Blickwinkel der Maschine.

1 Vgl. Charniak/McDermott 1987, 21ff. Obwohl die Abduktion an mehreren Stellen ihres Handbuchs *Introduction to Artificial Intelligence* aufgegriffen wird, sucht man den Namen Peirce vergeblich.

2 Vgl. Taranto 1983, 1984; Thagard 1986; Schoonefeld 1986.

3 Vgl. Winograd/Flores, *Understanding Computers and Cognition*, 1987, 63 u. 107f.

4 Vgl. Ketner, "Peirce and Turing" (1988) sowie Walthers früher Hinweis auf Peirces "existential graphs", die "heute als "Graphentheorie" allgemein bekannt ist und die im Bereich des Programmierens elektronischer Rechenanlagen und der abstrakten System- und Automatentheorie große Bedeutung gewonnen hat" (Walther 1967, 20).

Mir liegt hingegen daran zu zeigen, daß man auf der Grundlage der Peirce-Semiotik auch eine komplementäre Position vertreten kann. Informationstechnische Systeme lassen sich nicht nur besser nutzen, indem die Software "benutzerfreundlicher" und "intelligenter" wird. Dem sind ohnehin Grenzen gesetzt. Die Schwierigkeiten der MMK lassen sich auch durch verbesserte Qualifikation der Systembenutzer vermindern. Hier gibt es jedoch noch keine fachdidaktischen Arbeiten, die über einen Rezeptstatus "gestandener Praktiker" hinausgehen. Mit anderen Worten: Die Vermittlung informationstechnischer Kenntnisse muß verbessert werden. Hierzu bedarf es einer zuständigen "Didaktik der Informationsverarbeitung", die erst allmählich beginnt, Konturen anzunehmen.⁵ Eine solche Fachdidaktik sollte indes semiotisch fundiert sein. Dabei werden die komplexen Lernbedürfnisse des Systemanwenders als neue, ungewohnte Art der Zeichenrezeption, -formulierung und -manipulation interpretiert. Insbesondere die Arbeiten von Peirce halten geeignete erkenntnis-, sprach- und kommunikationstheoretische Grundlagen bereit, um den Gegenstand "MMK" zu analysieren und hieraus didaktische Schlüsse zu ziehen.

Die Mensch-Maschine-Kommunikation aus semiotischer Sicht

Die Besonderheiten der Mensch-Maschine-Kommunikation entstehen aus einem ungewohnten semiotischen Spannungsverhältnis. In der MMK stehen sich zwei Instanzen gegenüber, deren kleinster gemeinsamer Nenner darin besteht, Daten zu organisieren und Zeichen zu manipulieren. Ihre charakteristischen Unterschiede werden besonders deutlich, wenn man jeweils die Besonderheiten der Zeichenstruktur und Zeichenfunktionen bei maschineller und menschlicher Informationsverarbeitung betrachtet.

a) Informationstechnische Systeme

Die Software eines informationstechnischen Systems interpretiert jedes Mittel (bzw. Zeichenträger) als Legizeichen, dem eindeutig und ausschließlich eine bestimmte Funktion innerhalb des Programmsystems zugeordnet ist. Für das jeweilige Programm erfüllt das Legizeichen eine reine Identifizierungsfunktion, um hierauf eindeutig eine Programmroutine auszulösen und einen Zustand des Systems herbeizuführen. Die Programmiersprachen, auf deren Grundlage sämtliche Anwendungen (auch die Programmiersprachen selbst) beruhen, sind vollkommen formallogische Kunstsprachen, deren syntaktische Verknüpfungsmöglichkeiten und semantische Bezüge präzise definiert sind, einschließlich ihrer Freiheitsgrade im Funktionsprozeß.⁶ Sie weisen ein abgeschlossenes Zeichen-

⁵ Vgl. Peters (1990).

⁶ Natürlich kann jeder geübte Anwender mit einer formalen Sprache auch neue Funktionen/Befehle erstellen. Selbst ungeübte Anwender sind in der Lage, neue Funktionen durch Verkettungen von anderen Befehlen zu erstellen: Makros, einfache Batch-Routinen etc. - jedoch nur solche, die als gültige Kombinationsmöglichkeiten vorgesehen sind.

repertoire⁷ auf. Jede Instruktion an das Programm ist nur bei definierten Systemzuständen zugelassen, mit anderen Instruktionen vereinbar und führt deterministisch an dieser Programmstelle immer dieselbe Zustandsänderung herbei. Zusätzlich kann derselbe Befehl in verschiedenen Programmmuständen verschiedene Funktionen auslösen, doch immer auf dieselbe Art wiederholt werden.

b) Systembenutzer

Den von Systementwicklern determinierten Bedeutungsstrukturen des Programms stehen prinzipiell offene Sinnstrukturen der Benutzersemiosen gegenüber. Für den Anwender stellen sich die Programmcodierungen als Zeichen verschiedener Klassen dar: als Ikon (Piktogramme), (degenerierte⁸) Indizes (Warntöne) oder Symbole (Befehls**wörter**). In seinen Augen ist es deswegen weder beliebig, mit welchen Mitteln eine Funktion codiert ist, noch was er spontan unter der Bedeutung versteht, weil a) verschiedene Mittel unterschiedliche kommunikative Leistungsmöglichkeiten beinhalten und b) unterschiedliche Vorerfahrungen und Fähigkeiten von Anwendern mit diesen Mitteln zum Tragen kommen. Die Entscheidung für ein Mittel bei der Codierung während der Systementwicklung orientiert sich deshalb ausschließlich an der Sinnfälligkeit für die Anwender, d.h. an der **Verständlichkeit** der an der Schnittstelle angebotenen Zeichen. Sie mündet in der Frage für den Entwickler der Mensch-Maschine-Schnittstelle: "Welche Zeichen wähle ich für die Funktion aus, so daß sie am ehesten die **Vorstellungen** bzw. die Modelle des Benutzers über das System treffen?"⁹

Die Frage für den (semiotisch sensibilisierten) Didaktiker besteht nun darin, ausgehend vom vorgefundenen System die Verständnisschwierigkeiten der Anwender zu analysieren und überwinden zu helfen: Welche potentiellen Vorstellungen, Hypothesen, Heuristiken, Assoziationen oder Handlungspläne erzeugen die Sprach- und Bildzeichen des Programms? Welche sind selbsterklärend, welche potentiell irreführend; sind zusätzliche Analogien und Metaphern angebracht? Darüber hinaus ist es für den Systemvermittler schwer nachzuvollziehen, welche **Bedeutung der Anwender** denn jeweils den Programmzeichen entnimmt. Denn diese Bedeutung ist während der Einarbeitung in ein Programm für ihn keineswegs etwas Stabiles.

Im Peirceschen Zeichenmodell wird "Bedeutung" als Interpretanten-Relation verstanden. Einzelzeichen haben einen offenen, rhematischen Interpretanten. Das heißt: Zeichen "haben" keine feste Bedeutung, sondern diese wird jeweils neu in

7 Mit "Zeichen" sind sowohl Befehle und Kürzel kommando-orientierter Sprachen gemeint als auch die Bezeichnungen der ausgewählten Menüpunkte oder Piktogramme. Nur indirekt von dieser Problematik betroffen sind die Zeichen eines Programms, die Informationen über sich selbst bereithalten: Hilfsbildschirme, Text in Nachrichtefeldern, Fehlermeldungen etc.

8 "Degeneriert" bedeutet, abgeleitet, abgeschwächt zu sein.

9 "Der Systemdesigner ... muß entscheiden, inwieweit er seine Ausdrucksmittel dem Benutzer anpaßt, um ihm das System verständlich zu machen, und inwieweit er unbekannte Benennungen wählt, um den Benutzer auf die besonderen Eigenarten des Rechners hinzuweisen" (Herrmann 1986, 214).

einem Denk- oder Kommunikationsakt, einer Semiose, vom Zeichenbenutzer festgelegt: "Meaning, conventionally or not, does not "exist" hypotatically in signs, signs are used to create meaning dynamically, and "take on" meaning only in and by that process" (Robertson 1982, 185). Wir knüpfen im Denken immer schon an die Repräsentation anderer Sachverhalte in Zeichen an, die durch ihre soziale Verankerung virtuell viele Bezeichnungsrelationen zulassen. Dieselben Sachverhalte können hierdurch immer differenzierter beschrieben werden, weil sie verfügbar werden durch die Zuordnung eines Mittels, mit dem das Subjekt bereits Prädikationserfahrungen besitzt. Durch sie können sich Zeichenbenutzer in einer Semiose jederzeit eine neue Mittel-Objekt-Relation herstellen. Durch die Funktion des Interpretanten hat Peirce damit das Reflexivitätsmerkmal im Zeichenmodell verankert. Jeder Kommunikant ist durch die Struktur des Zeichens potentiell in der Lage, die ausgewählte Mittel-Objekt-Relation zu reflektieren. Je abstrakter die kommunizierten Sachverhalte aber sind, wie etwa in der MMK, desto notgedrungen präziser und kontrollierter wird die Reflexion über den Zeichengebrauch, um die Interpretationsvielfalt zu begrenzen. Gerade der Bedeutungsbegriff stellt die Flexibilität von Zeichen als Werkzeuge heraus, mit denen die jeweiligen Kommunikationsbedürfnisse bewältigt werden können.

Mit welchen Verständnisproblemen kämpft nun der Systembenutzer bei der Mensch-Maschine-Kommunikation?

Wichtig erscheint mir zunächst, den weiter oben angeführten Vorstellungsbegriff hervorzuheben, weil die interpretationsbedürftigen Funktionen selbst nur in ihrer Darstellung auf der Programmoberfläche der sinnlichen Wahrnehmung zugänglich sind.¹⁰ Die interne Zustandsveränderung im System kann, abstrakt wie sie ist, gar nicht das reale Objekt eines Programmzeichens für den Anwender sein. Eine extensionale Semantikkonzeption ist für das Zeichenverständnis des Anwenders demnach in den meisten Fällen¹¹ unangebracht: "Stattdessen steht der sprachliche Ausdruck einfach für den Endzustand unbekannter Prozeduren, die [zudem], weil sie so schnell ablaufen, ... gar nicht mehr wahrgenommen werden können" (Meder 1986, 711).

Doch diese anwenderorientierte Resultat-Semantik Meders erschöpft nicht das Bedeutungsspektrum für die Anwender. Das Ergebnis einer Prozedur mit ihrer Bedeutung gleichzusetzen, ist eher kennzeichnend für Anfänger. Auch für den Didaktiker ist es am Anfang einer Vermittlung zweckmäßig, die Frage "Was bedeutet X?" mit "Was tut X?" gleichzusetzen. Je nach Kenntnisstand kann sich die Bedeutung eines Befehls allerdings auch auf dessen Arbeitsweise und nicht nur

10 Dies ist der entscheidende Ansatzpunkt, Analogien und Metaphern bei der Vermittlung der Programmfunktionen eine wichtige Rolle zuzusprechen. Auf software-ergonomischer Seite ist man auf Erkenntnisse angewiesen, welche mentalen Modelle sich Computerbenutzer über das System machen, vgl. hierzu Johnson-Laird 1983.

11 Ein größeres Erklärungsbedürfnis über maschineninterne Arbeitsweisen tritt nach meiner Erfahrung erst bei Funktionen auf, die bislang als typisch menschliche Leistungen gelten, z.B. bei dem Befehl "Rechtschreibprüfung" oder insgesamt bei natürlichsprachlichen Eingabeformulierungen in Datenbanksystemen.

auf das Resultat beziehen, z.B. um die Ergebnisse abschätzen zu können.¹² Darüber hinaus ist jedes Zeichen durch seine Anwendungsvoraussetzungen und Verknüpfungsmöglichkeiten Teil eines Superzeichens, eine Repräsentation von komplexen Handlungsintentionen und -plänen.¹³ Hier verläßt die Semantik die reine "Wort"-Ebene. Welche Konsequenzen haben diese unterschiedlichen Zeichen- und die drei Bedeutungsstrukturen (extensionale, Resultat- und Superzeichen-Semantik) in der Informationsverarbeitung von Mensch und Maschine nun für Anwender?

Verständnis- und Lernschwierigkeiten des Systembenutzers

Für den Kommunikant in der Mensch-Maschine-Dyade sind die Mittel der Systemausgabe, aber auch die selbst eingegebenen Befehle merkwürdig vertraut und gleichsam fremd. Er findet Wörter, Wortfolgen oder Piktogramme vor, die er zum Teil aus zwischenmenschlichen Kommunikationsformen oder aus Handlungs- bzw. Lebenszusammenhängen (Papierkorb) kennt. Dies ist natürlich die besondere Leistung der Software-Entwicklung. Man kann sich unter vielen Befehlszeichen spontan etwas vorstellen, weil sie sich auf analoge Verwendungen in der natürlichen Sprache stützen. Üblicherweise benutzt und versteht ein Anwender sie, um mit ihnen seine Umwelt handelnd oder sprachlich zu bewältigen. Mit ihnen verbindet er oft eine lange Erfahrung in Verwendungssituationen. Gleichzeitig ist ihm aber das syntaktische und semantische Potential dieser Zeichen als Programmzeichen nicht klar. Sie begegnen ihm in der Lernphase als Regulativ: Entweder er kommuniziert eindeutig, fehlerfrei und zielangemessen, oder das System reagiert nicht bzw. falsch.¹⁴

Zu Beginn der Lernphase besteht lediglich ein "umgangssprachliches" Vorwissen über jene Zeichen, denen man in der MMK begegnet und deren Bedeutungspotential man bereits aus der Welterfahrung oder aus der Erfahrung mit anderen Programmen kennt. Hierzu gesellen sich mehr oder weniger diffuse Vorstellungen über Funktionen des Systems, bei denen Benutzer geneigt sind, sie mit früheren Arbeitsmitteln zu vergleichen.

Zu jenem Zeitpunkt kann der Anwender nur Hypothesen über die richtige Anwendung der Programmzeichen anstellen. Seine Interpretationen sind noch sehr vage, da er zum Beispiel den erforderlichen Systemstatus nicht kennt, der über die Korrektheit der Eingabe entscheidet. Es ist für ihn also schwierig, die Ver-

- 12 Zum Beispiel ist es erforderlich, die Funktionsweise einer Rechtschreibprüfung (Zeichenvergleich etc.) zu verstehen, um nachzuvollziehen, warum manche Fehler nicht vom Programm entdeckt wurden.
- 13 Ein Benutzer kann verschiedene syntaktische Verknüpfungen von Programmzeichen auslösen und die gleichen Funktionsergebnisse erzielen: vgl. z.B. die verschiedenen Ebenen der Formateinstellung bei MS-WORD.
- 14 Dieser Umstand grenzt die MMK auch größtenteils vom Fremdsprachenlernen ab, wo a) die Mittel unbekannt sind und b) wie in jeder natürlichen Sprache das semantische Spektrum ungleich größer ist als in formalen Systemen.

wendungssituation des Programmzeichens richtig zu identifizieren ("muß der Cursor hier stehen, bevor ich ...?").

Darüber hinaus besteht ein Lernbedarf des Anwenders abzuwägen, ob die von ihm gewählten Zeichen semantisch dem entsprechen, was er will, und wie er sie als Befehl auslöst. Dies setzt allerdings die Fähigkeit voraus, inhaltliche Aufgaben/Probleme überhaupt in einer Form zu repräsentieren, die es erlaubt, diese zu strukturieren: "One problem is the user awareness of tasks ..." (Sutcliffe/Old 1987, 39). Des weiteren besteht das Problem, die Handlungsintentionen vorzuformulieren und in entsprechende Befehlsfolgen umzusetzen: Dies betrifft zum einen die natürliche Sprache: "Reflexion auf die eigene Sprache darf man dem normalen Sprecher ... kaum unterstellen" (Lieb 1987, 69). Zum anderen betrifft es die Syntax und Semantik angewandter Programm-Befehle oder konzeptionelle Vorstellungen von Aufgabenstellungen und Handlungsplänen: "Users are certainly unaware of their own user models ..." (Sutcliffe/Old 1987, 39).

Ohne Instruktionen über das System sind für den Anwender sehr viele Eingabevarianten denkbar. Jede fehlerhafte Aktion mit dem Programm schränkt indes die unkorrekten Vermutungen über die Programmzeichen ein, so daß die semantische Konstruktionsregel, -idee oder -gewohnheit immer wieder, aber zunehmend weniger auf seine Angemessenheit geprüft werden muß.

Die Vermutungen, die während der Lernphase über die Programmzeichen gebildet werden, standardisieren deren Bedeutung immer stärker. Wird die Funktion einwandfrei beherrscht, so ist der Lernprozeß für die betreffenden Zeichen (folgen) bis auf den Kontext zu anderen Funktionen (Superzeichen) abgeschlossen. Dabei ist der Anwender gezwungen, die Hypothesen unter zwei Aspekten zu kontrollieren. Seine wachsenden MMK-Kenntnisse erweitern und differenzieren sich in bezug a) auf Funktionswissen und Zeichenverwendung im Begriffssystem des jeweiligen Software-Produkts¹⁵, b) auf die eigenen natürlichsprachlichen Erfahrungen mit den Mitteln und c) auf Erfahrungen mit denselben Programmzeichen in anderen Software-Produkten¹⁶. Hierdurch werden jedoch Bedeutungsinterferenzen ausgelöst, die den Anwender teilweise vor erhebliche Probleme stellen.

Denkt man bei dem Befehlswort "Löschen" an einen Kassettenrecorder, so vermutet man einen unwiederbringlichen physikalischen Datenverlust. Da bei den meisten Programmen das "Löschen" einer Datei sich lediglich auf ihren Eintrag in einem Dateiverzeichnis bezieht, die Daten also mit speziellen Programmen "gerettet" werden können, wird eine falsche Hypothese gebildet. Sieht man dagegen einen Papierkorb, so weiß man noch nicht, daß die dort "hineingeworfenen" Daten möglicherweise beliebig oft abgerufen, d.h. mehrfach aus dem Papierkorb herausgeholt werden können (was definitiv nicht der Erfahrung mit Papierkörben

15 Hier muß ein Didaktiker beachten: Ähnliche "Benennungen mit unterschiedlicher Bedeutung werden verwechselt" (Herrmann 1986, 92).

16 Vgl. Peters 1990.

entspricht). Mit anderen Worten: Die Hypothese über die Bedeutung des Zeichens ist unvollständig. Darüber hinaus vermutet man (natürlich), alle in den Papierkorb "geworfenen" Daten verblieben dort und würden nicht vom nächsten "Abfall" vernichtet: Der dritte Typ fehlerhafter Vermutungen spricht dem Zeichen mehr semantische Merkmale zu als es besitzt.¹⁷ Der Interpretant ist also am Anfang der Lernphase unvollständig oder mit fehlerhaften Hypothesen besetzt. Die wahrgenommenen Sinzeichen des Programms werden noch nicht als regelhafte Verkörperung des Legizeichens interpretiert.

Über die Programmzeichen gelangt der Anwender also zu falschen Vorstellungen über deren Objekt. Das reale Objekt eines Befehlszeichens, die vollständigen syntaktischen und semantischen Merkmale einer Funktion, die je nach Kontext den neuen Programm- und Datenzustand als Verarbeitungsergebnis beinhaltet, wird noch nicht dem Mittel zugeordnet. Die aktuelle Auffassung/Idee von der richtigen Zeichenverwendung des Anwenders (in bezug auf den Verarbeitungsprozeß, das Funktionsergebnis etc.), weicht hiervon ab.

Demgegenüber ist in der Phase des eingearbeiteten Zustandes das Objekt in seinen semantischen Merkmalen bekannt.¹⁸ Dies beinhaltet auch das Wissen um seine Einbindung als Superzeichen, als Wissen über Anwendungsvoraussetzungen und Verknüpfungsmöglichkeiten des Zeichens. Die Interpretation des Zeichens wird vom Interpreten vollständig determiniert. Die Zeichenverwendung wird durch zunehmende Häufigkeit ähnlich habitualisiert wie in der mündlichen Kommunikation. Die jeweilige Auffassung von dem syntaktischen und semantischen Funktionsumfang fällt letztlich ("in the long use") mit dem realen Objekt des Programmzeichens, d.h. dem real implementierten Funktionsumfang zusammen. Für einfache Aufgabenbearbeitungen reduzieren sich also die Anwendungssemiosen von einem "habit change" zur Aktualisierung von "habits".

Aus semiotischer Sicht ist es deshalb erforderlich, den Anwender zur wirkungsvollen Reflexion über die Programmzeichen im System zu befähigen. Hierfür ist die unmittelbare Rückkopplung des Programms ein ständiger Anlaß. Der Anwender muß also mit Techniken intrapersonaler Kommunikation vertraut gemacht werden. Diese Techniken sind eine elementare Grundlage für selbstgesteuertes Lernen. Sie entsprechen zum einen (sprach-) logischen Operationen wie abstrahieren, systematisieren, vergleichen, induzieren usw., zum anderen der Technik, Handlungen zu planen und Ergebnisse zu kontrollieren sowie Hypothesen zu bilden und sie zu bewerten.

17 Diese Formen der Fehlinterpretation sind m.E. für einen nennenswerten Teil unvollständiger oder inkorrektur "mentaler Modelle" verantwortlich. Da es aber ohnehin fraglich ist, ob und wie solche Modelle gebildet werden und strukturiert sind, ist es für einen Didaktiker zweckmäßiger, unmittelbar bei den einzelnen Zeicheninterpretationen anzusetzen.

18 Natürlich kann ein Anwender, wenn er dem Programmzeichen das reale Objekt zuordnen kann, a) an den Funktionsprozeß des Befehls denken, b) an Verarbeitungsergebnisse und c) an Verweise auf andere Befehle in Handlungskontexten (Superzeichen). Mit anderen Worten: die Anwender abstrahieren je nach Interpretant von bestimmten Aspekten des realen Objekts.

Anstatt diese methodischen Konsequenzen hier ausführlich zu erörtern, möchte ich mich darauf beschränken, stellvertretend mit zwei Zitaten auf den Fundus hinzuweisen, den das Werk von Peirce Didaktikern bereitstellt. Sie verdeutlichen meines Erachtens, wie aktuell viele seiner Schriften nach wie vor sind. Beide Zitate liefern präzise Erklärungen zu der Frage, wie man über die Steuerung der Aufmerksamkeit und Hypothesenbildung des Lernenden bei der Systemvermittlung Regelwissen erarbeiten kann.

Attention is roused when the same phenomenon presents itself repeatedly on different occasions, or the same predicate in different subjects. We see that A has a certain character, that B has the same, C has the same: and this excites our attention, so that we say "These have this character." Thus attention is an act of induction; ... a habit arises, when, having had the sensations of performing a certain act, *m*, on several occasions *a, b, c*, we come to do it upon every occurrence of the general event *I*, of which *a, b* and *c* are special cases ... Thus the formation of a habit is an induction, and is therefore necessarily connected with attention or abstraction" (Peirce 1931-58, CP 5.296f.).

What should an explanatory hypothesis be to be worthy to rank as a hypothesis? Of course, it must explain the facts. But what other conditions ought to be fulfilled to be good? ... Its end is, through subjection to the test of experiment, to lead to the avoidance of all surprise and to the establishment of a habit of positive expectation that shall not be disappointed ... Of course, this maxim of abduction [= hypothesis] supposes that, as people say, we "are to believe only what we actually see" ... (Peirce 1931-58, CP 5.197f.).

Der erste Teil des Aufsatzes erörterte die grundlegenden Unterschiede in der Informationsverarbeitung bei Mensch und Maschine. Im zweiten Teil möchte ich nun den Anwendungsbezug von Peirce an einem Problembereich der MMK konkretisieren und beispielhaft einige didaktische Folgerungen aus der semiotischen Analyse ziehen.

Sprachliche und bildhafte Repräsentationen in der Mensch-Maschine-Kommunikation

Ein Didaktiker wird beim Thema "Informationsverarbeitung" in zwei Problemereichen mit Besonderheiten bildhafter Informationen konfrontiert. Neben dem Einsatz von Analogien bei der Vermittlung (oder bei der Erstellung von Lernsystemen) spielen "Bilder" vor allem eine Rolle bei der Vermittlung piktogramm-orientierter Software-Produkte. Hier stellt sich die Frage, inwieweit Lehr-Lern-Prozesse beeinflusst werden von den Unterschieden im Vergleich zu sprachlicher Informationsverarbeitung. Ich werde daher zunächst terminologisch und inhaltlich klären, welche Funktionen bildhafte Mittel bei der computergestützten Informationsverarbeitung erfüllen.

Der Terminus "Bild" ist natürlich zu vage, als daß er die in seinem Kontext auftretenden semiotischen Lehr-Lern-Prozesse hinreichend differenziert erfaßt. Auch die Bezeichnung "nicht-sprachliche Darstellung" reicht hier nicht aus, weil sie zu allgemein ist und zudem bildhafte Aspekte der natürlichen Sprache unterschlägt. Wesentlich exakter ist dagegen die Peircesche Differenzierung des Objektbezugs in der Zeichentriade, mit der man analysieren kann, durch welche Beziehungen ein Mittel auf das Objekt verweist. Peirce definiert das ikonische Zeichen als eine Relation, nach der Mittel auf Objekte durch mindestens ein gemeinsames Merkmal verweisen. In der Semiose wird diese als Ähnlichkeitsrelation von dem Zeichenbenutzer wiedererkannt bzw. explizit interpretiert.

Der Peircesche Ikon-Begriff (vgl. CP 1.379) enthält sowohl homologe als auch analoge Relationen zwischen Mittel und Objekt. Deren gemeinsame Merkmale können gegenständlich-abbildend übereinstimmen (z.B. Photos, manche Piktogramme), gemeinsame nicht-qualitative Relationen aufweisen (z.B. Fieberkurven, Landkarten) oder in abstrakter Beziehung zueinander stehen (z.B. religiöse Symbole, Metaphern, vgl. CP 5.66).

Hieraus ergibt sich zunächst ein sehr weites Spektrum möglicher Zeichenbeziehungen, die jeden Fall *"erlebter Ähnlichkeit"* von Zeichen und Bezeichnetem [umfassen], wie diese von einem Subjekt festgestellt [werden], unabhängig davon, was immer man in metrischer Hinsicht unter "ähnlich" verstehen mag" (Walcher 1981, 13). Für Kommunikationsanalysen ist der Ikon-Begriff trotzdem überaus hilfreich, weil mit ihm - gerade auch für Software-Piktogramme - geprüft werden kann, "welche Merkmalsträger jeweils welche funktionale Bedeutung erlangen" (ebd.). Dabei ist er aufgrund der im folgenden beschriebenen Charakteristika gut geeignet, individuell unterschiedliche Lernschwierigkeiten offenzulegen.

Zum einen sind ikonische Zeichen rhematisch (Einzelzeichen), d.h. sie sind keine Sätze (nicht wahrheitsfähig). Ikons können zudem nicht gut miteinander verknüpft werden, weil sie keine diskreten Elemente aufweisen (wie etwa Grapheme für Sprachen). Zum anderen ist ihre Bedeutungskonstitution durch einen Zeicheninterpret, wie in der Sprache, keineswegs eindeutig - in der Regel nicht wegen ihres Abstraktionsgrades: "Im Gegensatz zu den konventionalisierten Willkürzeichen sämtlicher symbolisch fixierter Sprachen handelt es sich hierbei im allgemeinen um nicht-willkürliche, nicht- oder partial-konventionalisierte anschaulich-topologische Gebilde" (Walcher 1981, 17).

Da ein ikonisches Zeichen mangels expliziter semantischer Relationen nicht "von sich aus sagt", welche Elemente als Primär- und welche als Sekundärmerkmale anzusehen sind, kann sich der Interpret auf eine Vielzahl von Merkmalen des Ikons beziehen - je nachdem, welche qualitativen Merkmale bei Mittel und Objekt gerade als ähnlich empfunden werden. Die Vagheit der Interpretation nimmt prinzipiell ab, je konkreter-gegenständlicher die Merkmale des Mittels sind: "Für

Benutzer ... kann jedoch eine solche detailtreue, realistische Darstellung zu Erkenntnisschwierigkeiten führen, da sie einige der dargestellten Details nicht mit dem dazugehörigen Objekt assoziieren" (Staufer 1987, 9). Zudem ist auch bei stärker "abbildenden" Relationen nicht sichergestellt, auf welches Merkmal der Interpret sich gerade bezieht. Unter Hinweis auf die angelsächsische Terminologie gilt es also zu beachten, daß dasselbe "picture" unterschiedliche "images" (Vorstellungen) erzeugen kann.¹⁹ Es bedarf einer verbindlichen Setzung, einer festen Bedeutungszuweisung, damit die Interpretationsbreite von ikonischen Zeichen wirkungsvoll eingeschränkt werden kann. Diese Setzung erfolgt zwangsläufig bei (Software-) Piktogrammen, deren Aufgabe es u.a. ist, so verständlich bzw. eindeutig für die Befehle zu stehen wie möglich. Piktogramme werden eingesetzt, um "der menschlichen Art zu handeln und zu denken stärker entgegenzukommen" (v. Benda, Vorwort in Staufer 1987). Hierbei sind die spezifischen Merkmale der piktogramm-orientierten Schnittstellen, inklusive ihrer Stärken und Schwächen, zu beachten.

Semiotisch gesehen, sind Piktogramme ikonische Zeichen.²⁰ Die formallogisch determinierte Bedeutung von Software-Piktogrammen ist gegenüber dem realen Objekt, der Programmfunktion, abgeschwächt und in ihren Merkmalen aus funktionalen Ähnlichkeiten (vgl. das "Papierkorb"-Beispiel) abgeleitet. Piktogramme in Computersystemen sind also genauso arbiträr wie sprachliche Programmzeichen; sie müssen eindeutig sein und sind nur durch geeignetes Vorwissen über ihre Bedeutung interpretierbar.

Jedoch besitzen Piktogramme spezifische kommunikative Charakteristika. Auf dem Bildschirm bezeichnen sie in der Regel Datenobjekte (wie Dateien/Verzeichnisse) und Programmfunktionen, die in Menüform dargeboten werden. Im Vergleich zu sprachlichen Zeichen wird besonders ihre Anschaulichkeit betont, obwohl man gegenüber anderen ikonischen Zeichen wie Bildern, Filmen etc. einen größeren Abstraktions- und Standardisierungsanteil hervorhebt - wobei der Abstraktionsgrad eine gewisse Spannweite besitzt (detailtreu bis typisierend²¹).

Die Kategorie "Anschaulichkeit" bei ikonischen Zeichen verweist auf gemeinsame Merkmale mit dem - meist gegenständlichen - Objekt: Hat ein Strichmännchen,

19 In der Kognitionspsychologie ist es bislang umstritten, ob mentale Bilder eher abbildend, mit propositionalen/begrifflichem Gehalt oder in einem "dual coding"-Prozeß im Gedächtnis repräsentiert werden; vgl. Paivio 1977, dagegen Kosslyn 1980.

20 "Interessant ist die etymologische Bedeutung des Terminus [von "Piktogramm"]. Die wörtliche Übertragung "geschriebenes Bild" deutet darauf hin, daß Piktogramme sowohl Eigenschaften von schriftlicher Informationsvermittlung als auch von bildhafter Darstellung aufweisen. Piktogramme stellen tatsächlich in gewisser Weise ein Bindeglied dar, welches zwischen den Polen alphanumerischer und bildhafter Informationsübertragung steht" (Staufer 1987, 5). Diese als Definition verstandene Beschreibung ist für die erste deutschsprachige Monographie zum Thema "Piktogramme für Computer" bedauerlicherweise recht allgemein, zumal sie die Art des ikonischen Erkenntnisprozesses nicht hinreichend klärt.

21 Typisierungen sind wichtig, da gut dokumentiert ist, daß ein für Menschen "typisches Objekt" kein realer Gegenstand ist, der dann für die Objektklasse steht, sondern eine Zusammenstellung der als prototypisch erlebten Eigenschaften einzelner Objekte der Klasse (Vgl. Staufer 1987, 61f.).

das nur für das Merkmal "männlich" stehen soll, noch einen Bart, Hose etc.? Im allgemeinen bewirkt eine steigende Ikonizität (größere Merkmalsübereinstimmung) eine bessere Wahrnehmbarkeit und bessere Behaltensleistung des ikonischen Zeichens. In bezug auf die Programmfunktion ist eine gegenständliche Merkmalsübereinstimmung jedoch nur schwer möglich. Piktogramme als Befehlszeichen zeichnen sich durch **funktional ähnliche** Merkmale aus, die meistens im Mittel auf analoge Tätigkeitsbereiche verweisen (z.B. "Posteingangskorb")²².

Piktogramme sind also durch funktionale Analogien und typisierende Darstellung unter Umständen verständlicher als sprachliche Befehlszeichen. Wie das Papierkorb/Lösch-Beispiel zeigt, kann man erst nach einem **funktionalen** Merkmalsvergleich ("Was tut, und wie benutze ich Funktion X?") der Mittel entscheiden, welche (wahrscheinlich) ausgelösten Semiosen dem Programmbefehl eher entsprechen. Durch die gegenständlich-funktionale Ähnlichkeitsrelation bestehen also keine durchgängigen Vorteile von Piktogrammen gegenüber sprachlichen Befehlen. Ihre Stärke haben Piktogramme vor allem durch:

- unmittelbare visuelle Präsenz, die eine direkte Wahrnehmung gewährleistet (gegenüber sequentieller Informationsaufnahme wie bei Sprache) und das Gedächtnis/die Aufmerksamkeit des Anwenders entlastet;
- Berücksichtigung des großen Anteils visueller Informationsaufnahme beim Menschen;
- leichte Erinnerbarkeit bzw. hohen Wiedererkennungswert²³ wegen ihrer großen Verarbeitungstiefe der Information;
- sinnliche Anreicherung: keine "visuelle Monotonie";
- spielerische, motivierende Effekte, die Angst abbauen helfen;
- Förderung der Antizipation und Kontrolle des Werkzeugs, die modellhafte Annäherung an vertraute Handlungszusammenhänge ermöglicht: besonders bei adäquaten gegenständlichen Analogien ("Radiergummi, Pinsel, Farbtopf") sind sie unmittelbar verständlich; deshalb können sie als besonders benutzerfreundlich gelten, wenn die ehemals traditionell bewältigten Arbeiten ebenfalls handlungs- und gegenstands-

22 Etwas völlig anderes ist es natürlich, wenn die ikonischen Zeichen tatsächlich materiale oder relationale Eigenschaften mit dem repräsentierten Objekt gemeinsam haben, z.B. ein Leiterplattenentwurf bei CAD-Systemen als topologisches Ikon. Diese Simulationen würde man aber nicht als Piktogramme klassifizieren.

23 Auch wenn ein Piktogramm zunächst ebenso erklärungsbedürftig ist wie ein sprachliches Befehlszeichen, so ist nach seiner einmaligen Erklärung mit einer ausgezeichneten Behaltensleistung des Anwenders zu rechnen: "Ein Benutzer berichtete über ähnliche Erfahrungen, als er zum erstenmal das Piktogramm für "Papierkorb" bei dem Computer "Apple Lisa" sah. Nachdem er darauf hingewiesen wurde, daß das ihm zusammenhanglos erscheinende Symbol einen Papierkorb darstellen sollte, erkannte er daraufhin das Piktogramm immer als Abbild eines Papierkorbs" (Staufer 1987, 48).

orientiert waren, bei der die Tätigkeiten nicht allgemein repräsentiert bzw. kommuniziert werden mußten.²⁴

Diesen beachtlichen Vorteilen ikonischer Informationsdarstellung stehen allerdings auch Mängel gegenüber:

- Abstrakte sprachliche Informationen (z.B. ein Befehl wie "Übertragen" oder die Negationskategorie bei "Undo")²⁵ sind in der Regel nicht durch Piktogramme abbildbar, weil sie rein sprachlich repräsentiert werden und keine "ähnlichen" Entsprechungen in anderen Handlungszusammenhängen haben. Gerade die Zeichen eines Software-systems sind indes besonders abstrakt (vgl. Kolars 1969). Die Darstellungsproblematik betrifft dabei noch nicht einmal Funktionen und Abläufe. Auch weniger abstrakte Gegenstände wie Festplatten sind kaum "sinnfällig" darstellbar und daher verstärkt erklärungsbedürftig. Für eine Wahrnehmungsanalogie haben sie zu wenig markante Merkmale; das funktionale Merkmal "Speichermedium" läßt keine "bildhaften" Entsprechungen zu.
- Durch die mangelnde Syntax sind Piktogramme nicht verknüpfbar oder hierarchisierbar und damit vergleichsweise unflexibel. Differenziertere und längere Handlungsfolgen sind sprachlich einfacher zu bewältigen ("Bewege-Zeile-Von-Nach"), wenn ihre Linearität einleuchtend ist. Folgerichtig besteht die Konsequenz piktogrammorientierter Systeme auch darin, nach "Anklicken" des Zeichens weitere **sprachliche** Rückfragen zu stellen. Piktogramme enthalten deshalb weniger mächtige und komplexe Befehle.
- Trotz Nähe zu analogen Handlungskontexten bleiben die exakten Programmmerkmale (Wie löscht der Papierkorb genau?) weiteren Lernprozessen vorbehalten. Sie sind nicht unmittelbar im Mittel repräsentiert, das in dieser Hinsicht unbestimmt bleibt.
- Die rezeptive Informationsaufnahme wird zwar durch Piktogramme besser unterstützt; Menschen kommunizieren in Arbeitskontexten jedoch nur selten aktiv in Bildern. Piktogramme sind kaum in ihrer Abwesenheit formulierbar und verknüpfbar. Hiermit geht zwangsläufig eine gewisse Passivität einher, die charakteristisch ist für rechnerorientierte Informationsformen wie Menüs.

24 Stauffer nennt darüber hinaus eine Reihe von Benutzermerkmalen, denen Piktogramme in "ihrer leichten kognitiven Verarbeitbarkeit" (1987, 38) entgegenkommen: höheres Alter (Merkfähigkeit), Anfänger (Führung, Angstabbau), Gelegenheitsbenutzer (Erinnerbarkeit, Einfachheit), Statusdenken (Abkehr von klassischer Textverarbeitung), Vorwissen (schnelle Erlernbarkeit), gute allgemeine Intelligenz/Problemlösungsfähigkeit (Visualisierung), Strukturiertheit der Aufgabe (geringere Flexibilität).

25 Die Probleme der Veranschaulichung/bildlichen Umsetzung **logischer** Sachverhalte kann man sehr gut an Peirces "existential graphs" (1983, 171) demonstrieren.

- Berücksichtigt man den Komplexitätsgrad verschiedener Informationsverarbeitungsprozesse in Büros, erscheint es mir fraglich, ob Piktogramme tatsächlich an "mental Modellen der Büro-/Arbeitswelt" anknüpfen, die zusätzlich individuell verschieden sein können.
- Intrapersonale Kommunikationsprozesse haben zwar bildhaft-assoziative Anteile; für die bei der computergestützten Informationsverarbeitung notwendige Klarheit der repräsentierten Problemstellung reicht die geringe Differenzierungskraft von Piktogrammen aber kaum aus. Dies betrifft besonders die wichtige Superzeichenbildung. Hier müssen die Relationen, Verknüpfungsmöglichkeiten und Seiteneffekte einzelner Programmzeichen exakt antizipiert werden.

Auch bei Piktogrammen bleiben die Lernprozesse des Anwenders unverändert. Daher bleiben auch die hiermit verbundenen Lernziele identisch. Während die Ausweitung der aktiven Handlungsautonomie durch Piktogramm-Menüs eingeschränkt wird, bestehen demgegenüber mehr Ansatzpunkte und Anregungen für spielerische, ästhetisch motivierte Handlungsintentionen des Benutzers. Piktogrammorientierte Bildschirmoberflächen erleichtern das Verständnis der abstrakten Funktionsprinzipien (Prozeß-Semantik), weil die visuellen Zeichen analog repräsentieren, was jeweils getan wird. Sie erschweren hingegen das Verständnis der Superzeichenbildung. Gegenüber diesen vergleichsweise geringen Unterschieden in den didaktischen Überlegungen lassen sich nun einige methodische Besonderheiten nennen.

Durch den hohen Motivationswert und leichte Einarbeitung in piktogrammorientierte Benutzeroberflächen kann der Unterrichtsablauf in der Anfangsphase offener gestaltet werden. Die Exploration der Anwender sollte gezielt herausgefordert werden. Nach nur wenigen Vorgaben von Basiskonzepten sammeln die Teilnehmer zunächst eigene Programmerkahrungen. Man kann annehmen, daß die Anwender dabei eine Reihe von Vermutungen/Hypothesen über die Arbeitsweise des Programms bilden, die sich leichter durch induktive Vermittlungsstrategien aufgreifen lassen. Die Phasen der selbständigen Arbeit können eher verlängert werden, da die Orientierung am Bildschirm, die damit verbundene Handlungskontrolle und der Umgang mit Fehlern einfacher sein dürften. Allerdings sollte durch die schlechte Verknüpfbarkeit der ikonischen Programmzeichen sichergestellt werden, daß die Benutzer zur Stufe der Superzeichenbildung geführt werden. Darüber hinaus sollten die funktionellen Merkmale eines Piktogramms während der Vermittlung auch sprachlich repräsentiert werden, um vage Interpretationsprozesse abzufangen und semiotische Differenzierungen zu fördern.

BIBLIOGRAPHIE

- Charniak, E.; McDermott, D. [1987]: *Introduction to Artificial Intelligence*. Reading/Mass. usw.
- Dreyfus, H.L. [1985]: *Die Grenzen künstlicher Intelligenz: was Computer nicht können*. Königstein/Ts.
- Herrmann, Th. [1986]: *Zur Gestaltung der Mensch-Computer-Interaktion - Systemerklärung als kommunikatives Problem*. Tübingen
- Johnson-Laird, P.N. [1983]: *Mental Models*. Cambridge
- Ketner, K.L. [1988]: Peirce and Turing: Comparisons and conjectures. In: *Semiotica* 1,2 (1988) 33-61
- Kolers, P.A. [1969]: Some formal characteristics of piktogramms. In: *American Scientist* 3 (1969) 348-363
- Kosslyn, S.M. [1980]: *Image and mind*. Cambridge
- Lieb, H.-H. [1987]: Sprache und Intentionalität: der Zusammenbruch des Kognitivismus. In: *Sprachtheorie - Der Sprachbegriff in Wissenschaft und Alltag. Jahrbuch 1986 des Instituts für Deutsche Sprache* (hrsg. v. R. Wimmer), 11-16. Düsseldorf
- Meder, N. [1986]: Superzeichensemantik oder: Der Sprachspieler in möglichen Welten. In: *Pädagogische Rundschau* 40 (1986) 705-718
- Paivio, A. [1977]: Images, propositions, and knowledge. In: *Images, perception, and knowledge* (ed. by J.M. Nicholas), 47-72. Dordrecht
- Peirce, Ch.S. [1931]: *Collected Papers*. Ed. by Ch. Hartshorne, P. Weiss (Bde. 1-2 [1932], 3-4 [1933], 5-6 [1935]) und A.W. Burks (Bde. 7-8 [1958]). Cambridge/Mass.
- Peirce, Ch.S. [1973]: *Lectures on Pragmatism. Vorlesungen über Pragmatismus* (hrsg. v. E. Walther). Hamburg
- Peirce, Ch.S. [1983]: *Phänomen und Logik der Zeichen* (hrsg. v. H. Pape). Frankfurt a.M.
- Peirce, Ch.S. [1986]: *Semiotische Schriften*. Band 1 (hrsg. u. übersetzt v. C. Kloesel; H. Pape). Frankfurt a.M.
- Peters, W. [1990]: *Didaktik der Informationsverarbeitung aus sprachdidaktischer Sicht*. Tübingen (in Vorbereitung)
- Robertson, I.T. [1982]: Individual Differences in Information Processing Strategy and Style. In: *IEE International Conference on Man-Machine-Systems*. Wembley
- Schoonfeld, C.H. [1986]: A Sign-theoretical Model for Semantic Structure in Language. In: *COLING Proceedings, 11th International Conference on Computer Linguistics. Bonn Aug. 25th to 29th 1986*. Bonn
- Staufer, M. [1987]: *Piktogramme für Computer*. Berlin/New York
- Sutcliffe, A.G.; Old, A.C. [1987]: Do users know they have user models? Some experiences in the practice of user modelling. In: *Human-Computer Interaction - INTERACT '87. Proceedings of the Second IFIP Conference on HCI* (ed. by H.-J. Bullinger; B. Shackel), 35-42. Amsterdam usw.
- Taranto, R.E. [1983]: Semiotics of Artificial Intelligence. In: *Semiosis* 3 (1983) 7-21
- Taranto, R.E. [1984]: Artificial Intelligence And Expert Systems. In: *Semiosis* 3 (1984) 38-41
- Thagard, P. [1986]: Charles Peirce, Sherlock Holmes and Artificial Intelligence. In: *Semiotica* 3,4 (1986) 289-295
- Walcher, K.-P. [1981]: Zur Definition des Bildes. In: *Zeichenkonstitution* (hrsg. v. A. Lange-Seidl), 12-17. Berlin/New York

Walther, E. [1967]: Vorwort und Einleitung (I und II). In: Ch.S. Peirce, *Die Festigung der Überzeugung*, T-41. Baden-Baden

Winograd, T.; Flores, F. [1988]: *Understanding Computers and Cognition. A New Foundation for Design*. 3. Aufl. Reading/Mass. usw.

SUMMARY

In this essay, man-machine-communication is indicated as new range of application of the semiotics of Peirce. Hereby, the mediation of information-technical systems emerges from a didactical viewpoint of language into the foreground of semiotic analysis. In a first step, the differences of sign-structure and sign-functions are described in the face of mechanical and human information processing as learning problems of the system user. Then, one of the relevant problems in man-machine-communication, i.e. the communicative efficiency of iconic signs, is didactically expounded in an exemplary manner.

SEMIOSIS

55
56

Internationale Zeitschrift
für Semiotik und Ästhetik
14. Jahrgang, Heft 3/4, 1989

INHALT

Cornelie Leopold:	Anmerkungen zum Dualitätsprinzip in Geometrie und Semiotik	3
Karl Gfesser:	Bemerkungen zum "Zeichenband"	17
Jorge Bogarin:	Für wen ist etwas ein Zeichen?	31
Regina Claussen:	Zeichen und Ideologie - Vom ideologiekritischen Wert der Rhetorik	39
Udo Bayer:	"Der Zipfel einer Welt" - Übergänge zwischen Objektthematik und ästhetischer Eigenrealität	47
Matthias Götz:	Wo ist der Ausgang? Wenn Bilder Auskunft geben: Piktogramme	59
Wolfram Peters:	Die Peirce-Semiotik als Ansatzpunkt für eine Didaktik der Informationsverarbeitung	71
Alfred Toth:	"Es war einmal ein alter König, der hatte eine Tochter, die war die schönste Jungfrau auf der Welt." Pragmasyntaktische Oberflächen- und fundamental-kategoriale Tiefenstrukturen im Rahmen einer semiotischen Linguistik	87
<i>Charles Sanders Peirce Sesquicentennial International Congress 5.-10. September 1989 an der Harvard University, Cambridge/Massachusetts (Cornelie Leopold, Karl Gfesser)</i>		103
<i>Rapport succinct sur la manifestation (Joëlle Réthoré)</i>		107
Thomasz Komendziński:	Peirce in Poland. Complete Polish Bibliography of Charles S. Peirce 1897-1987	109
Mitteilung des Herausgebers und der Redaktion		119
Inhalt von Jahrgang 14		121