

# Informatik als Grundbildung

H. Wedekind · E. Ortner · R. Inhetveen

**Der Einfluss der Informatik auf unseren Alltag ist enorm. Der Einfluss der Informatik auf die Lehrpläne unserer allgemeinbildenden Schulen ist gering. Man mag das mit der Jugendlichkeit des Faches zu begründen versuchen und einen beliebten Vergleich mit der Mathematik anstreben, die heute ein wesentlicher Teil der Grundbildung ist. Was die Terminologie anbetrifft, so lehnen wir uns eng an das Deutsche PISA-Konsortium an [1].**

klärung. „Ein Mensch gilt als gebildet,“ heißt es noch bei J.G. Fichte 1808 in seinen berühmten Reden an die deutsche Nation „wenn er ganzheitlich aus sich selbst heraus will, was allgemein und sittlich erfordert ist“. Das idealistische Bildungsziel „Sittlichkeit“ eines Fichte löst heute nur noch ein Schmunzeln aus. Was von Fichte und der Aufklärung aber geblieben ist, ist die Forderung nach Selbsttätigkeit. „Aus sich selbst heraus“ muss etwas geschehen, was heute oftmals vergessen wird. Zum Selbsttätigwerden bedarf es aber einer Orientierung mit dem dazugehörigen Wissen (Mittelstraß). Bloßes Verfügungswissen wird dem zugewiesen, was wir Ausbildung („education“) nennen. Wenn wir Terhart in seiner Nach-PISA-Studie folgen ([6], S. 22), ist Sittlichkeit heute „durch individuelle Handlungsfähigkeit zum Lösen von Problemen in

Grundbildung, um die es uns geht, vermittelt eine Basisqualifikation und ist nicht mit Allgemeinbildung zu verwechseln. Es geht wie in der PISA-Studie um Basiskompetenzen, die Voraussetzung für den Erwerb einer Allgemeinbildung sind. Allgemeinbildung ist durch einen kaum auszumachenden Horizont begrenzt und steht deshalb nicht zur Debatte.

Bildung ist ein Zentralbegriff der Auf-

der modernen Welt“ zu ersetzen. Es geht nach Terhart in der Grundbildung um „funktionale Kompetenzen der Welterschließung und Problembewältigung“. Und damit sind wir sehr nahe am anglo-amerikanischen Begriff „literacy“, der sich von dem einfachen Lese/Schreibvermögen zu dem entwickelt hat, was im Deutschen seit den Zeiten der Aufklärung „Bildung“ heißt. „Basic literacy“ (Grundbildung) steht speziellen Bildungsformen wie z.B. der „computer literacy“ gegenüber. Das Programm „Informatik in den Schulen“ hat sich dieser Form angenommen. Rechneraufbau, Algorithmen, Programmiersprachen und alles rund um den PC stehen im Mittelpunkt eines Unterrichts, für den in der Regel sogar ein eigenes Schulfach vorgesehen ist. Es geht uns nicht um das Problem „computer literacy“ oder Informatik in den Schulen. Es geht uns um das Thema „Informatik als Grundbildung“. Gäbe es die Informatik als Grundbildung schon, dann würde sich diese Arbeit erübrigen. Die Frage, die von uns zu beantworten ist, lautet: *Was muss in einer Informatik-Grundbildung methodisch vorangestellt werden, damit ein spezielles Bildungsangebot im Sinne einer „computer literacy“ überhaupt verstanden wird? Es geht uns um eine logische Propädeutik, wie sie von Kamlah u. Lorenzen [2] entwickelt wurde, zugeschnitten*

---

DOI 10.1007/s00287-004-0383-z  
© Springer-Verlag 2004

Prof. Dr. H. Wedekind  
Universität Erlangen-Nürnberg,  
Institut für Informatik,  
E-Mail: wedekind@informatik.uni-erlangen.de

E. Ortner · R. Inhetveen  
TU Darmstadt,  
Fachbereich Recht- und Wirtschaftswissenschaften

auf das Fach Informatik, das viele andere Fächer penetriert hat.

Die Beantwortung dieser Frage ist nicht einfach und bedarf einer längeren Darstellung, wenn wir nicht in einer Allgemeinheit stecken bleiben wollen. Es werden ohne Anspruch auf Vollständigkeit sechs Lehrstücke beschrieben, die alle eine entscheidende Rolle spielen – nicht nur in einer dann folgenden Informatik an Schulen und Hochschulen. Andere Fächer, die konstruktiv, d.h. schrittweise, zirkelfrei und alles explizit machend gelehrt werden, hängen in gleicher Weise von diesen Lehrstücken ab. Die sechs Themen, die im Folgenden nur skizzenhaft dargestellt werden, um dann in weiteren Folgen auf sie näher einzugehen, lauten:

- Schema und Ausprägung,
- Bildung von Elementarsätzen,
- Gleichheit und Abstraktion,
- Objektsprache / Metasprache,
- Namensgebung und Kennzeichnung,
- Logik und Geltungssicherung von Behauptungen.

Es handelt sich um einen sechs-dimensionalen Raum, dessen Hauptkoordinaten abzustecken sind. Dieser Raum hat auch eine zeitliche Tiefe, die in dieser Aufzählung verloren geht. Als Ersatz für diese siebte Dimension mag die sechste gelten, seit der Aufklärung die problematischste in unserem Geschäft.

Wir sind uns bewusst, dass eine Forderung „Informatik als Grundbildung“ für viele eine Ungeheuerlichkeit darstellt und deshalb hoffentlich zu einer regen Diskussion Anlass gibt. Im Sinne der PISA-Studie [1], die drei Grundbildungsformen kennt („reading literacy“, „mathematical literacy“ und „natural science literacy“), soll eine vierte Form, eine „informatical literacy“ ins Leben gerufen werden. Dabei sei es völlig unbenommen, in welchem Rahmen der Lehrstoff dargeboten wird. Der Mathematik-Unterricht ist hierfür prima facie genauso geeignet wie der Deutsch-Unterricht. Man kann auch an den Informatik-Unterricht denken, falls dieser sich als geeignet erweisen sollte.

Ein Hauptanliegen dieses Beitrags betrifft auch das Thema „Geschlechterunterschiede und Basis-kompetenzen“, ein zentrales Kapitel der PISA-Studie [1]. Es ist auffallend, dass Mädchen in Sachen Sprachkompetenz („reading literacy“) international gegenüber den Jungen eine hohe Domi-

nanz aufweisen. In der Mathematik liegt der Fall umgekehrt, nur nicht ganz so drastisch. Die darzustellende Informatik-Grundbildung ist invariant bezüglich des Antagonismus „Sprache vs. Mathematik“, oder krasser „weiblich vs. männlich“. Fundamental-wissenschaftlich, besser gesagt, philosophisch gesehen, ist diese Unterscheidung auch ohne Sinn. Sie löst sich, dialektisch gesprochen, in der Synthese einer Informatik-Grundbildung auf. Das ist natürlich eine Hypothese und eine Hoffnung, und somit vorerst eine Spekulation. An einem empirischen Nachweis im Sinne der PISA-Forschung ist nicht vorbeizukommen. Aber erst muss etwas vorhanden sein, bevor man es testet. Dabei ist klar, dass eine Aufbereitung zu einem vermittelbaren Lehrstoff (Didaktik) eine entscheidende Rolle spielt.

Hier nun sechs Kurzfassungen.

### Schema und Ausprägung

Das Begriffspaar „Schema – Ausprägung“ spielt in der Informatik, aber auch in anderen Wissenschaften eine zentrale Rolle. Das Paar wird aber in der Informatik in aller Strenge thematisiert. *Programme* als Schemata haben *Kontrollfäden* („threads of control“) als Ausprägungen. *Classes* sind Schemata und *Objekte* ihre Ausprägungen. *Relationen* in Datenbanken sind konzeptionelle Schemata und haben *Tupeln* als Ausprägungen, das Klicken auf einem Bildschirm folgt einem Zeigehandlungsschema usw. Je nach Sprachmodell und Autor spricht man allgemeiner auch von „competence“ (Schema) und „performance“ (Ausprägung) (**Chomsky**), von „langue“ (Schema) und „parole“ (Ausprägung) (**de Saussure**) und von „type“ (Schema) und „token“ (Ausprägung) (**Peirce**). Schemata sind universelle Beschreibungen von Gegenständen unsere Welt. Ausprägungen sind singular. Bildung kann als Schemaerwerb aufgefasst werden, natürlich unter der Maßgabe der Selbsttätigkeit („self-acquisition“).

### Bildung von Elementarsätzen

„Fido ist ein Hund“ oder „Heute ist Donnerstag“ sind Elementarsätze, die erst in der Logik zu komplexen Sätzen verknüpft werden. Das Schema eines Elementarsatzes lautet:  $N \in P$  mit  $N$  als einem Namen, allgemein Nominator genannt, und  $P$  als einem Prädikator. Es wird nicht Prädikat gesagt, um einer Verwechslung mit dem Terminus „Prädikat“

einer empirischen Grammatik vorzubeugen. Das Zeichen ‚ε‘ heißt die Kopula und wird im Deutschen durch das vieldeutige Wort ‚ist‘ wiedergegeben. Eine Hauptaufgabe ist, diese Vieldeutigkeit herauszuarbeiten. An Elementarsätzen wird die wichtige Unterscheidung von Extension (Umfang) und Intension (Inhalt) eines Prädikators eingeübt.

## Gleichheit und Abstraktion

Eine explizite Darstellung des Abstraktionsvorgangs ist ein Kernstück der Informatik-Grundbildung. Wenn in einem Text „Margarine“ durch „Butter“ ersetzt werden darf, ohne die Geltung des Textes zu verändern, dann liegt eine Abstraktion von Prädikaten zu einem Begriff vor. Abstraktion baut auf der Spezifikation einer Äquivalenzrelation auf. Im Falle von „Margarine“ und „Butter“ spricht man bei sprachlicher Gleichbehandlung von Synonymität. Berühmte Abstraktionen wie: Von den Ziffern (Zählzeichen) zur Zahl, von Folgen zu Mengen, von Termen zu Funktionen betreffen zwar die mathematische Grundbildung. Dass jede gute Schnittstelle eine Abstraktion ist im Hinblick auf äquivalent zu betrachtende Implementierungen, das ist eine Einsicht, die früh zu vermitteln ist. Es sollte deutlich gemacht werden, dass die Abstraktion beide Grundbildungen, die mathematische und die informatische miteinander verbindet.

In der Informatik-Grundbildung sollte mit höchster Genauigkeit gearbeitet werden. Ungenauigkeit verursacht Unsicherheit. Deshalb ist in diesem Kapitel der Begriff „Information“ präzise als Abstraktion einzuführen.

## Objektsprache/Metasprache

Wenn Gegenstände einer Sprache selbst zum Gegenstand des Sprechens werden, ist diese Sprache gesondert als Metasprache auszuzeichnen. Die objektsprachliche Aussage *Peter ist kurz* soll bedeuten, dass ein Individuum mit dem Namen Peter von kleinem Wuchs ist. Auf die Ebene einer Metasprache gehoben, muss geschrieben werden: „*Peter*“ ist kurz. Gemeint ist: Das Wort, besser das Wortschema „Peter“ ist kurz. Wortschemata, oder gar ganze Satz schemata, sind die Objekte einer Metasprache und müssen als solche durch Anführungszeichen hervorgehoben werden, um Missverständnisse und Paradoxien zu vermeiden. Der Satz „*kurz*“ ist kurz macht Sinn und zeigt das wichtige Phänomen einer Selbstbeschreibung (Autologie)

auf. Der Satz *kurz ist kurz* macht hingegen keinen Sinn. Wichtig ist herauszustellen, dass Schemata als Namen auftreten. Der zugewiesene Prädikator *kurz* ist objektsprachlich ein Objekt-Prädikator, metasprachlich aber ein Meta-Prädikator. Das Reden über Sprache ist typisch für die Informatik, die dann Meta-Informatik genannt werden kann. Dieses Phänomen gibt es drüben bei Sprache und Mathematik als Grundbildung auch, nicht aber in den Naturwissenschaften, die objektsprachlich verhaftet bleiben. Die empirische Grammatik z.B. des Deutschen oder des Lateinischen ist eine Sprachbeschreibung und eine metasprachliche Grundlage, auf die in der informatischen Grundbildung zurückgegriffen werden sollte.

## Namensgebung und Kennzeichnung

„Die Hauptstadt Frankreichs“ ist eine echte Kennzeichnung, weil ein Objekt, das existiert und eindeutig ist, beschrieben wird und mit dem Eigennamen Paris versehen werden kann. Jede URL ist eine echte Kennzeichnung. Existenz, aber nicht Eindeutigkeit ist häufig im praktischen Leben gegeben. Man spricht dann von potentiellen Kennzeichnungen, wenn Eindeutigkeit nicht garantiert werden kann. Die standesamtlichen Festdaten wie Name, Geburtstag und -ort etc. sind streng genommen nur eine potentielle Kennzeichnung. Das Aktenzeichen, das beim Standesamt vergeben wird, macht den Registriervorgang aber eindeutig. In der Informatik sind echte und potentielle Kennzeichnungen gang und gäbe. Ein Zusammenbauteil wird zum Beispiel durch seine Komponenten echt oder potentiell gekennzeichnet. Wichtig ist, die große Bedeutung der Kennzeichnung in unserem Leben herauszuarbeiten. Leben wir doch in einer nahezu namenlosen Welt. Geographische und astronomische Objekte tragen zuweilen einen Eigennamen. Hinzu kommen noch die Lebewesen Mensch und manchmal auch Tier. Zu allen anderen Objekten muss zwecks Referenzierung eine Kennzeichnung aufgebaut werden, die genauso behandelt werden kann wie Eigennamen.

## Logik und Geltungssicherung von Behauptungen

Man sagt, die Logik sei kurz, leicht und alt. Das Prädikat „kurz“ mag stimmen, „leicht“ ist mehr als ein Irrtum, der dazu geführt hat, Logik aus dem Unterricht seit Wilhelm von Humboldt zu verban-

nen. Das Prädikat „alt“ stimmt. Die Logik ist rund 2500 Jahre alt. Sie wurde aber eingeführt, um den Dialog um zusammengesetzte Elementarsätze, die behauptet werden, zu steuern. Das Wort „Kontrolllogik“ ist, so gesehen, ein Pleonasmus, ein „weißer Schimmel“. Wer behauptet, so lautet der Grundsatz der Logik mit einer tiefen erzieherischen Wirkung, der muss auch Belege herbeischaffen, um zu zeigen, dass das, was ausgesagt wird, auch stimmt. Es geht um Geltung. Man kann auch das stark belastete Wort „Wahrheit“ benutzen. Eine Wahrheit fällt nicht vom Himmel. Sie muss in der Regel schwer erarbeitet werden. Logik in einer informatischen Grundbildung darf nicht verwechselt werden mit einer Logik, die heute die Theorie der Informatik bestimmt. Die Verwechslung von Theorie und Grundlagen ist ein großes Missverständnis des Faches, das beseitigt werden sollte. In dem bekannten Lexikon der Philosophie von Mauthner steht unter dem Stichwort „Logik der Tatsachen“ die eindringliche Mahnung: „Nicht nur Hegel, ..., hat sich der Sünde schuldig gemacht, Logik außerhalb der Sprache zu suchen“.

## Teil I: Schema und Ausprägung

Gliederung:

- 1 Schema und Ausprägung als Gegenstände einer Rede.
- 2 Schemaerwerb und Bildung.
- 3 Schemaerwerb und Gerechtigkeit

### 1. Schema und Ausprägung als Gegenstände einer Rede

Dem Begriffspaar „Schema – Ausprägung“ ist in der Informatik eine dominante Stellung zugewiesen worden.

Rechner-Programme werden als Schema eines Ablaufs oder Vorgangs aufgefasst. Zur Laufzeit eines Programmschemas werden Ausprägungen („instances“) erzeugt. Der Gedanke des zeitlichen Ablaufs einer Ausprägung wird auch durch den Begriff „Faden“ bzw. „Ablauffaden“ („thread“, „thread of control“) wiedergegeben [4]. Der Bewegungsablauf eines Roboterarms ist eine Ausprägung eines Programms, das als Schema vorhanden ist. Für Datenbanksysteme werden konzeptionelle Schemata häufig als relationale Schemata entwickelt. Tupeln oder Zeilen, die zu einer Relation gehören sind ihre Ausprägungen. Eine Lehrer-Relation oder eine Lehrer-Tabelle ist, so gesehen,

ein Schema, die einzelnen Zeilen, für jeden Lehrer genau eine, sind die Ausprägungen dieser Relation. Abstrakte Datentypen sind wie alle Typen Schemata. Betrachten wir z.B. den Abstrakten Datentyp „Quadrat“, objektorientiert dargestellt durch eine *Class* mit einer Methode zum Zeichnen von Linien. Ein besonderes Quadrat, das z.B. herausgezeichnet wird, ist eine Ausprägung, die zuweilen auch *Objekt* einer *Class* „Quadrat“ genannt wird. Alle einfachen Datentypen der Programmiersprachen sind Schemata. Der Typ *Integer* z.B. ist ein Schema für die Ausprägungen 1, 2, 3, ..., n.

Die Beispiele zeigen, dass das Begriffspaar „Schema-Ausprägung“ zum Kern der Informatik gehört. Aus diesem Grunde ist auch berechtigt vorgeschlagen worden, das Fach *In.forma.tik* mit seinem Basiswort *Form*, vom Lateinischen *forma*, durch *In.schema.tik* zu ersetzen. Denn das griechische Wort *schema* (σχῆμα) entspricht dem lateinischen Wort *forma*. Der Begriff „Form“ wird in den Wissenschaften sehr vielschichtig benutzt. Man stellt z.B. gerne Form und Inhalt (Materie) gegenüber. Logiker sprechen von Schlüssen kraft der Form, also von Gültigkeiten, die allein schon aus formalen Gründen gesichert sind. Man könnte aber auch von Schlüssen kraft des Schemas und von schematischen Gründen sprechen. Es wäre wünschenswert, wenn der Formbegriff durch den Schemabegriff expliziert würde. Man weiß dann sofort, dass es nicht um Geometrie und den dort dominanten Formbegriff geht. Insofern ist das Wort *Inschematik* ernst, und nicht nur als ein kleiner Scherz am Rande aufzufassen.

Was ist nun in allgemeiner Sprache ein Schema und was seine Ausprägungen? Von K. Lorenz [4] stammt die folgende, einfache Definition: Ein Schema stellt einen universellen (allgemeinen), eine Ausprägung einen singulären (besonderen) Aspekt eines Gegenstands einer Rede dar. Das Wort *Gegenstand* wie auch das aus dem Lateinischen stammende Fremdwort *Objekt* steht für nichts Geheimnisvolles. Gegenstände kommen in einer menschlichen Rede vor. „Nur wo menschliche Rede ist, werden Gegenstände von anderen Gegenständen unterschieden“, heißt es in der logischen Propädeutik von Kamlah/Lorenzen [2] ganz ohne ontologisches Pathos. So wie wir sagen „Dieser Gegenstand ist ein Hammer“, also das Wort *Gegenstand* als Verlängerung des Demonstrativpronomens *dies* benutzen, wollen wir sagen „dieser Gegenstand ist ein Sche-

ma“, wenn wir z.B. das Schema einer Datenbank meinen, oder wir sagen „dieser Gegenstand ist eine Ausprägung“, wenn ein Tupel oder eine Zeile einer Relation bzw. Tabelle gemeint ist. Statt „dies ist ein P“, wobei P für Hammer, Schema, Ausprägung oder für was immer stehen mag, redet man aus bloß syntaktischen Gründen in der Form (Schema) „dieser Gegenstand ist ein P“, ohne dass der Inhalt verändert wird.

Das Begriffspaar „Schema – Ausprägung“ hat eine lange Tradition in der Geschichte der Sprachlehre, wobei unterschiedliche Wörter benutzt wurden, je nachdem, welcher Rahmen um die vorgetragene Sprachlehre gelegt wurde. Der berühmte amerikanische Linguist N. Chomsky (geb. 1928) nannte das Begriffspaar „competence – performance“. Mit einem Schema ist Kompetenz verbunden, eine Ausprägung zu vollziehen („to perform“). Statt Ausprägung werden in allgemeineren Sprachrahmen auch die Wörter *Aktualisierung* bzw. Instantiierung („occurrence“ und „instantiation“) benutzt. Weiter zurück in der Geschichte ist der Schweizer Linguist F. de Saussure (1857–1913) zu nennen, der das Begriffspaar „langue – parole“ (Sprache-Rede) prägte. Eine ganze Sprache, gemeint ist die Grammatik einer Sprache plus ihrem Wortschatz, wurde von de Saussure als ein universeller Aspekt, als ein Schema gesehen, unter dem tatsächliche Rede auch in beliebigen Wiederholungen ausgeführt werden kann. Noch weiter zurück ist der amerikanische Logiker C.S. Peirce (1839–1914) zu erwähnen, der die Wörter „typen – token“ insbesondere im Hinblick auf Symbole einführte. Den schematische Aspekt eines Symbols nannte er „type“, seine Aktualisierung z.B. in einem hingeschriebenen Zeichen „token“. Seit den Zeiten von Peirce, und nicht erst seit dem Aufkommen moderner Programmiersprachen, wird *integer* ein Typ und die Zeichen 1, 2, 3, ..., n Token oder Ausprägungen genannt.

Es war für die moderne Weiterentwicklung des Begriffspaares von großer Bedeutung, Schema und Ausprägung in die Lehre von der Natur des Menschen, in eine Anthropologie sagt man gelehrt, einzubeziehen [4]. K. Lorenz (geb. 1932) ist dies gelungen, indem er von der Elementarsituation eines Dialogs zwischen einem aktiven Sprecher und einem passiven Hörer ausging. Vom Sprecher, also von der Person, die eine Sprechhandlung vollzieht („to perform“), wird gesagt, dass sie eine Ausprä-

gung erzeugt. Vom Hörer, also der Person, für die es die Sprechhandlung passiv gibt, wird gesagt, dass sie ein Schema der Handlung als etwas Allgemeines erkennt, um die Sprechhandlung zu verstehen. Der Sprecher produziert eine Ausprägung nach einem Schema, das der Hörer verfügbar haben muss, um die Ausprägung zu verstehen. Beide, Sprecher und Hörer, haben ein Schema gemeinsam, der eine, um zu erzeugen („to perform“), der andere, um zu erkennen („to recognize“) und zu verstehen („to understand“). Sie treten ein in eine Kommunikation (vom Lateinischen *communis*: gemeinsam).

Eine Schreiber/Leser-Situation ist ähnlich der Situation zwischen einem Sprecher und einem Hörer. Wir wollen eine Schreiber/Leser-Situation darstellen, um ein Nicht-Verstehen zu exemplifizieren. Ein Schreiber möge die für einen Leser unlesbare Zeichenfolge *Haus* an eine Tafel schreiben. Die Kreidespuren auf der Tafel sind keine Ausprägung des Zeichenschemas *Haus*, weil das Schema für den Leser elementar nicht erkennbar, nicht lesbar ist. Unlesbarkeit ist als Phänomen allseits bekannt. Nehmen wir das Beispiel der chinesischen Schriftzeichen (Piktogramme). Europäer können die Zeichen im allgemeinen nicht lesen, weil ihnen die Schemata nicht verfügbar sind. In einer Gesellschaft, die sich Wissensgesellschaft nennt, müsste man sagen „Die Europäer *wissen* nicht, mit Piktogrammen umzugehen“. Statt *können* ist heute *wissen* gefragt. „Die Grammatik des Wortes *wissen* ist offenbar eng verwandt der Grammatik der Worte *können*, *imstande sein*. Aber auch eng verwandt der des Wortes *verstehen*. (Eine Technik beherrschen)“ sagte der Philosoph Ludwig Wittgenstein im § 150 seiner philosophischen Untersuchungen. Von dem einflussreichen Wittgenstein wird gleich unten nochmals die Rede sein. Er ermahnt uns aber mit dem Wort *wissen* sorgfältig umzugehen. Man stelle sich vor, das Wort *Könnensgesellschaft* würde demnächst eingeführt, weil *können* ja sehr nah bei *wissen* liegt. Ob die Gesellschaft, in der so geredet wird, dann endlich merkt, wie albern das alles klingt?

Es ist heute üblich, Schemata von drei verschiedenen Standpunkten aus zu betrachten. Man nennt sie pragmatisch, semantisch und syntaktisch. Pragmatisch heißt ein Standpunkt, wenn danach gefragt wird, was man mit einem Schema anfangen kann. Semantisch heißt der Aspekt, wenn nach der Bedeutung gefragt wird, und von Syntax ist die Rede,

wenn zur Debatte steht, ob das Schema auch korrekt, d.h. regelgerecht (man sagt auch wohlgeformt) gebildet wurde. Eine ganzheitliche Sicht stellt sich auf alle drei Standpunkte. Ein großes methodische Problem entsteht aber, wenn die Reihenfolge des Betrachtens auszuwählen ist: pragmatisch, semantisch, syntaktisch (Reihenfolge I), oder umgekehrt: syntaktisch, semantisch, pragmatisch (Reihenfolge II).

Die Frage, welche der Reihenfolgen auszuwählen ist, spaltet die Geister, nicht nur in der Informatik. Die Anhänger von Reihenfolge I nennen sich Pragmatiker. „Sprechen ist Handeln“ (vom Griechischen  $\pi\rho\alpha\zeta\iota\varsigma$ , praxis) ist die Hauptthese eines pragmatischen Vorgehens. Wenn wir sprechen, erzeugen wir handelnd Ausprägungen eines Schemas. Aus Handlungen lassen sich Bedeutungen ableiten. „Die Bedeutung eines Wortes ist sein Gebrauch in der Sprache“ lautet ein berühmter Satz des für die Informatik so wichtigen Philosophen Wittgenstein (1889–1951). Der Satz ist in seinen philosophischen Untersuchungen unter § 43 zu finden. Wittgenstein ist der Mann, der eine sprachpragmatische Wende eingeleitet hat. Seine Wirkung auf alle Wissenschaften, die sich mit Sprache befassen, war enorm. Wenn die aus einer Sprechhandlung erwachsene Bedeutung klar ist, bedarf das Ergebnis in einem weiteren Schritt einer wohlgeformten Darstellung, einer Syntax. Man darf auch von Orthographie sprechen.

Wenn Pragmatiker in dieser Reihenfolge eine Rangordnung der Probleme erblicken, geraten sie mit anderen Geistern in der Informatik in Streit, die die Reihenfolge II favorisieren. Wir nennen sie Entwickler. Bitte schön, werden die Anhänger der Reihenfolge II sagen, hier ist mein Schema, mein Rechnerprogramm. Es läuft noch nicht, es produziert noch keine Ausprägungen, weil noch einige syntaktische Fehler drin sind. Der Rechner nimmt syntaktische Fehler sehr übel und versagt seine Dienste. Aber das ist für die Herrn Pragmatiker nur drittrangig. Ob die Semantik den Spezifikationen, den festgelegten Bedeutungen entspricht, kann ohne ein laufendes Programm nicht festgestellt werden. Das ist aber ja auch nur zweitrangig für die Herrn Pragmatiker.

Pragmatiker und Entwickler leben in einem ironischen Verhältnis, obwohl beide wissen, dass sie voneinander abhängen. Der Pragmatiker erreicht nichts ohne eine syntaktisch und semantisch

saubere Entwicklung. Dem Entwickler droht die Belanglosigkeit, wenn es keinen Gebrauch für seine Schemata gibt. Die bekannte Ironie ist fehl am Platze. Sie entstammt nicht der Reihenfolge, sondern der Gewichtung der einzelnen Schritte. Der Satz „the most important part comes first“ ist wie bei jeder Zurücklegung eines Weges als Grundsatz falsch. Hochmütige Pragmatiker und hochmütige Entwickler mit Rangordnungen im Kopf sollten die Geschichte vom Lahmen und Blinden lesen, die alleine nichts erreichen, zusammen aber wenigstens etwas auf ihrem gemeinsamen Wege fertig bringen. Die Lehre über Reihenfolgen wird in den Wissenschaften auch Lehre vom Wege oder Methodologie genannt. Richtige und falsche Methodologien gibt es nicht in Wissenschaften, die sich kritisch nennen und die einem Dogmatismus abgeschworen haben. Es gibt nur begründete und unbegründete Methodologien, und jeder Wissenschaftler sollte wissen, dass Hochmut vor dem Fall kommt, den doch jeder gerne vermeiden möchte.

## 2. Schemaerwerb und Bildung

Mit den historischen und systematischen Bemerkungen im obigen Abschnitt soll auch auf einen wichtigen Zusammenhang beim Erwerb einer Grundbildung und beim Bildungserwerb überhaupt hingewiesen werden. Es ist zwar so, dass die Informatik das Thema „Schema – Ausprägung“ zu ihrem zentralen Anliegen gemacht hat, was von anderen Disziplinen nicht behauptet werden kann. Man könnte mit anderen Worten auch sagen: Die Informatik ist sich des großen Themas bewusst, das anderen aus dem Blickfeld geraten ist und nur unscharf am Rande wahrgenommen wird. Wie kommt das? Die Antwort auf diese Frage ist relativ einfach. Wer etwas auf den Rechner bringen will, was für uns lebensbedeutsam ist, der muss unsere Lebenswelt kritisch, häufig sogar ab ovo, rekonstruieren. Der Rückgriff auf so etwas Grundsätzliches wie die Natur des Menschen, was man Anthropologie nennt, ist dabei unerlässlich. Für die Physik und anderen Naturwissenschaften zum Beispiel ist die Rede von Schema und Ausprägung etwas Nebensächliches. „Das weiß man doch“ lautet die bekannte Antwort. Im Sprachunterricht der Schulen geht die Lehre von Schema und Ausprägung zum Leidwesen der Informatik wegen der Fülle des Stoffes in vielen Fällen einfach unter. Zwar lernt der Schüler im Englischunterricht brav das Schema „to suffer

from something“ (an etwas leiden) als infinite Verbform. Er lernt auch, dass „I suffered from measles“ (Ich litt an Masern) eine korrekt gebildete Ausprägung ist. Dass aber eine ganze moderne Wissenschaft, die wir *Inschematik* nannten, von dem, was da im Sprachunterricht bloß als Exempel vorgeführt wird, lebt, das erfährt der Schüler nicht. Wer erfährt schon im Deutschunterricht, dass der Satz „Ich gratuliere Dir zum 15. Geburtstag“ nur etwas Singuläres ist gegenüber dem allgemeinen Schema „Jemand gratuliert jemandem zu etwas“. Oder schauen wir auf den Mathematikunterricht. Zur Debatte mag der Satz stehen: Für alle  $x$ : Wenn  $x > 1$ , gilt auch  $x^2 > 1$ . Im allgemeinen sagt der Lehrer schon, dass man zur Erleichterung auch den großen Buchstaben  $X$  einführen darf, um zu schreiben: Wenn  $X > 1$ , gilt auch  $X^2 > 1$ , wobei er hinzufügt, dass  $X$  als fest, aber beliebig anzusehen ist. Hoffentlich sagt der Lehrer auch, das  $X$  ein schematischer Buchstabe ist, der nicht quantifiziert werden darf und keine Variable im üblichen Sinne ist. Hoffentlich verweist der Lehrer auf das Fach *Inschematik* mit dem Hinweis, dass man dort über Schematisierungen eine Menge mehr lernen kann, was für das spätere Leben von sehr, sehr großer Bedeutung ist. Unter glücklichen Umständen sagt der Lehrer sogar, wenn man über Eigenschaften von Schemata redet, wie etwa bei  $X > 1$  oder bei „to suffer from“ ist *zwölf-buchstabig*, dass man dann auf die Ebene einer Metasprache gerät, über die man nicht nur im Fach *Inschematik* sehr viel lernen kann. Auch im Deutschunterricht sollte eine Regel wie „Alle Substantiva werden am Anfang groß geschrieben“ (für alle  $S$  gilt,  $S$  ist großbuchstabig) als Satz einer Metasprache aufgefasst werden. Man nennt ein solches Regelwerk im allgemeinen Grammatik. Den Satz „schön“ ist ein Adjektiv, mag ein Grammatiklehrer an die Tafel schreiben. Hoffentlich vergisst er dabei nicht die Anführungszeichen, die aus einem Wort der deutschen Objektsprache („use language“, Gebrauchssprache) den Namen eines Satzes in einer Metasprache („mention language“, Erwähnungssprache) machen. Ohne Anführungszeichen, wie z.B. beim Satz: *Schön ist das Mädchen*, schriebe der Grammatiklehrer etwas höchst Zweifelhafes an die Tafel, weil dann nicht herausgestellt würde, dass das Wort „schön“ nur als ein Sprachkonstrukt, nur als ein Schema auch ein Adjektiv ist.

Was ist Bildung? Man kann zu diesem Thema viele geistreiche Bücher finden, die ganze Bibliotheken füllen. Wenn Bildung aber nicht mehr unbedingt etwas Sittliches bewirken soll, wie noch bei J.G. Fichte im Jahre 1808, sondern nach PISA heute „funktionale Kompetenz der Welterschließung und Problembewältigung“ (Terhart) zum Gegenstand hat, und – wie im Deutschen üblich – dann als Anglizismus unter dem Wort *literacy* firmiert (man könnte im Deutschen auch statt *toll collect Maut*- bzw. englisch-näher *Zolleinzug* sagen), ja dann sollte es auch erlaubt sein, auf die anthropologischen Konzepte „Schema – Ausprägung“ zurückgreifen zu dürfen, wobei in unserer scheinbar anglophilen Welt ohne Bedenken und Einschränkungen auch die Wörter „type – token“ (seit Peirce) oder „competence – performance“ (seit Chomsky) verwendet werden sollten. Gemeint ist immer der universale bzw. der singuläre Aspekt in Bezug auf einen Gegenstand. Ein Hauptanliegen des ersten Teils unserer sechsteiligen Darstellung über „Informatik als Grundbildung“ ist eine Fokussierung auf die Aussage: Bildung, das sind erworbene Schemata, plus die Fähigkeit zum Selbsterwerb (Fichte), plus Gedächtnisleistung. Gebildete Menschen haben durchweg ein gutes Langzeit-Gedächtnis, weshalb das Auswendiglernen von Gedichten eine vorzügliche Übung ist, was die heutigen Pädagogik vergessen zu haben scheint. Ohne ein Selbsttätigwerden reduziert sich Bildung auf Ausbildung („education“). Bildung ist ein Selbstläufer, wenn eine Orientierung gegeben ist. Ausbildung bedarf einer permanenten Auffrischung in Kursen mit Credit points als Belohnung in abfragbaren Einheiten. Man sagt im Deutschen „Ich bilde mich“ (Aktivform) und „Ich werde ausgebildet“ (Passivform). Man sagt nicht „Ich werde gebildet“ (Passivform) und „Ich bilde mich aus“ (Aktivform). Bildung wie Ausbildung haben aber das Verfügen über Schemata und den Schematerwerb gemein. Um einen höheren Bildungs- bzw. Ausbildungsstand zu erreichen, ist der Besuch von Schulen, Hochschulen und Universitäten unerlässlich. Lehrer werden wissenschaftlich ausgebildet, um Schemata und nicht Ausprägungen zu vermitteln. Schemata befähigen, Ausprägungen zu verstehen, nicht umgekehrt, obwohl in Lehr- und Lernsituationen häufig das Erlernen von Schemata durch ein Vorführen von Ausprägungen eingeübt werden kann. Schreiben und nicht ausgeprägtes, schönes Schreiben (Kalligraphie) wie in der zum

großen Teil schemafreien Kunst stehen zur Debatte. Es gibt, frei nach Kant, keine schöne Wissenschaft, es gibt nur schöne Kunst.

Es ist ein großer Fehler zu glauben, dass ein Schemaerwerb nur in einem institutionellen Rahmen von Schulen stattfinden kann. Lehr- und Lernsituationen sind in unserem Leben dominant und mindestens so bedeutsam wie Entscheidungs-, Muße-, Transport- oder Herstellsituationen. Nehmen wir als Beispiel einen Vater, der seinem Sohn das Schwimmen beibringen will [2]. Der Vater demonstriert Schwimmbewegungen als Ausprägungen im Trockenen, indem er Armzüge mit gleichzeitigem Beingrätschen vorführt. Der Sohn imitiert das Schema ebenfalls als Ausprägung. Ob er die Ausprägungen als Schema gelernt hat, versucht er dann mit einiger Überwindung im Nassen. Beispiele für nicht institutionalisierte Lehr- und Lernsituationen aus der Welt der Sprache sind in gebotener Kürze schwieriger vorzutragen. Nehmen wir einen Ortsunkundigen an, der Fußgänger nach einer bestimmten Straße im Ort befragt. Der Fußgänger beschreibt das Schema des Wegs zum gesuchten Ziel, nicht mehr. Vorführen wäre eine besondere Gunst, die der Ortsunkundige nicht erwarten kann. Der Ortsunkundige ist auf sich selbst gestellt. Ob er sein Ziel erreicht, ist fraglich. Deshalb ist der Erfolg eines Schema-Erlernens im Bereich der Sprache immer auch ein Problem der Kontrolle. Viel komplizierter als das Wegesuchen in einer fremden Stadt sind z.B. das Erlernen des Schemas eines Revisionspfades („audit trail“) einer Debitorenbuchhaltung oder das in den Griff bekommen des Schemas zum Justieren einer NC-Maschine.

Auf jeden Falle spielt eine Schemaerwerbsfähigkeit („schema acquisition capability“) der Menschen eine zentrale Rolle.

### 3. Schemaerwerb und Gerechtigkeit

Schemaerwerb im Bildungs- und Ausbildungsprozess erfordert je nach Art und Schwierigkeit unterschiedliche Fähigkeiten und ein vielfältiges Können. Was den Abruf menschlicher Fähigkeiten anbetrifft, gilt seit alters her der alte römische Rechtsgrundsatz: „Ultra posse nemo obligatur“ oder „Über sein Können hinaus ist niemand verpflichtet“. Wenn ein Nicht-Können vorliegt, dann ist auch ein Nicht-Sollen gegeben. Schematisch kann geschrieben werden:  $\neg A \rightarrow \neg B$  (1), wobei der Haken ( $\neg$ ) die Verneinung und die schematischen Buch-

staben A und B durch Können bzw. Sollen zu aktualisieren sind. Dieses römische Gerechtigkeitsprinzip gilt auch heute nach bald 2000 Jahren noch in der westlichen Welt, obwohl es sozialpolitisch gerne durch ein zumutbares Können oder psychologisch durch ein Können-Mögen verwässert wird. Logik ist viel ernster, als Sozialpolitiker oder Psychologen das vermuten. Die Ernsthaftigkeit des römischen Gerechtigkeitsgesetzes wird z.B. auch herausgestellt, wenn wir mit dem großen Philosophen I. Kant (1724–1804) den Umkehrschluss (man sagt in der Logik auch „Kontraposition“) auf (1) anwenden. Klassisch logisch gilt:

Aus  $\neg A \rightarrow \neg B$  (1) folgt  $B \rightarrow A$  (2) „Wenn Du sollst, dann kannst Du auch“ (2) wird aus (1) gefolgert. Kant formulierte „... der Mensch kann etwas, weil ihm bewusst ist, dass er es soll“ (Kritik der Praktischen Vernunft, A 54).

Wir wollen im Folgenden (1) das Sozialprinzip und (2) das Leistungsprinzip beim Schemaerwerb nennen. Es ist klar, dass nach (1) etwa Behinderte, die schon allein nach dem Augenschein nicht bildungsfähig sind, nicht verpflichtet werden können. Das ist aber ein problemloser Fall. Problematisch wird die Situation, wenn keine Evidenz, sondern ein Feststellungsverfahren in Gang gesetzt werden muss, um auf Nicht-Sollen zu erkennen. Wer trägt die Beweislast des Nicht-Könnens? Z.B. der Staat, der das Prinzip (1) vertritt, oder die Person, die in Frage steht. Wir werden im Teil VI unserer Darstellung „Logik als Geltungssicherung von Behauptungen“ erfahren, dass auch die Beweislastfrage logisch einwandfrei geklärt ist. In der dialogischen Logik lernt man, dass der Proponent, in unserem Fall der Staat, zu nichts verpflichtet wird, wenn der Opponent, in unserem Fall die betroffene Person, die Behauptung des Nicht-Könnens nicht nachweist. Einfach nur zu sagen: „Ich kann nicht“ reicht nicht aus. Eine Begründung der Behauptung muss her, damit ein „Nicht-Können“ Geltung erlangt. Man erkennt: Bei der Frage der Beweislast „hört im allgemeinen der Spass auf“ und es wird ernst, logisch ernst.

Das Leistungsprinzip (2), das in gleicher Weise dialogisch logisch behandelt werden kann, ist wegen der positiven (affirmativen) Formulierung für jedermann einfacher zu verstehen. „Wenn etwas verlangt wird, dann muss das auch erreichbar sein“, ansonsten liegt ein unbilliges Verlangen vor. Äußerst problematisch bei (2) aber ist sein Er-



schließen aus (1) mit Hilfe des Umkehrschlusses. Kant galt und gilt heute noch als „bewunderter Rigorist“. Er glaubte vor mehr als 200 Jahren noch, dass die Logik ein abgeschlossenes Fach und nicht mehr entwicklungsfähig sei. Da hatte sich der große Mann, der körperlich nur 1,59 m groß gewesen sein soll, aber gewaltig geirrt, wie das bei bedeutenden Leuten auch zuweilen vorkommt. Die gewaltige Entwicklung der Logik, die mit G. Frege (1848–1925) 100 Jahre später einsetzte, konnte Kant nicht voraussehen. Hätte er es, dann wäre diesem kritischen Mann sofort klar geworden, dass sein Umkehrschluss so ohne weiteres nicht gilt. Der Schluss: Aus  $\neg A \rightarrow \neg B$  folgt  $B \rightarrow A$  gilt nämlich konstruktiv-logisch nicht. Wir werden im Teil VI noch sehen, dass die klassische Logik, die Kant noch benutzte, den berühmten Satz vom ausgeschlossenen Dritten (tertium non datur) als Annahme zugrunde legt. Wird dieser Satz aufgegeben, dann gilt der Umkehrschluss mit dem Schema, das Kant benutzte, nicht. Wenn wir den Satz vom ausgeschlossenen Dritten, also  $A \vee \neg A$ , erweitern und auch noch den häufig lebensweltlich anzutreffenden Fall des „es ist noch nicht entschieden, ob  $A \vee \neg A$  gilt“, die Römer sagten dazu *non liquet* (es fließt noch nicht) zulassen, dann entsteht eine viel allgemeinere Logik, die konstruktiv (intuitionistisch sagt man auch) genannt wird. Und dass etwas noch nicht fließt, das kommt häufig vor, insbesondere, wenn man an menschliche Arbeit denkt. Unser Problem ist, dass der Satz „Können  $\vee \neg$  Können ein Drittes gibt es nicht“ eine grobe und unzulässige Vereinfachung unserer Lebenswelt ist. Häufig, gerade im Ausbildungsbereich, ist ein Können noch im

Werden, es ist noch nicht vorhanden, aber es mag eine Hoffnung bestehen, dass es bald einsetzt. Wenn das so ist, gilt der Umkehrschluss nicht, was Kant noch nicht wusste. Hochinteressant ist natürlich die Tatsache dass die Umkehrung, also von (2) nach (1), auch konstruktiv gilt. Aus (2) kann (1) konstruktiv wie klassisch erschlossen werden. Konstruktiv-logisch gesprochen ist (2) stärker als (1). Aus diesem Grunde könnte der Satz „Du kannst etwas, weil Dir bewusst ist, dass Du es sollst“ im Emblem einer jeden Schule und Hochschule stehen.

Mit diesem abschließenden Eingehen auf „Logik und Geltungssicherung von Behauptungen“ wird zum Abschluss des Teils I, der als Öffnung einer Klammer aufzufassen ist, auf das Schließen dieser Klammer mit Teil VI verwiesen. Zuvor haben wir uns aber im nächsten Beitrag „Bildung von Elementarsätzen“ mit dem wichtigen Thema „Was ist eine rationale Grammatik?“ auseinander zu setzen. Eine empirische Grammatik wird im Sprachunterricht schon sehr früh eingeführt, z.B. wenn gelernt wird, dass zwischen Haupt- und Beiwörtern zu unterscheiden ist.

## Literatur

1. Deutsches PISA-Konsortium (Hrsg.): PISA 2000 – Basiskompetenzen von Schülerinnen und Schülern im internationalen Vergleich. Opladen: Leske + Budrich 2001
2. Kamlah, W., Lorenzen, P.: Logische Propädeutik, Vorschule des vernünftigen Redens. 3. Aufl. Stuttgart: J. B. Metzler 1996
3. Janich, P.: Logisch-pragmatische Propädeutik. Ein Grundkurs im philosophischen Reflektieren. Weilerswist: Velbrück Wissenschaft 2001
4. Lorenz, K.: Einführung in die philosophische Anthropologie. Darmstadt: Wissenschaftliche Buchgesellschaft 1990
5. Rechenberg, P., Pomberger G. (Hrsg.): Informatik Handbuch, 3. Aufl. München: Hanser 2002
6. Terhart, E.: Nach PISA – Bildungsqualität entwickeln. Hamburg: Europäische Verlagsanstalt 2002