

# Umsetzung von Grundkonzepten der Informatik zur fachlichen Orientierung im Informatikunterricht

Ludger Humbert

Universität Dortmund - Informatik XII  
Didaktik der Informatik

D-44221 Dortmund  
humbert@ls12.cs.uni-dortmund.de

23. August 2000

## Zusammenfassung

Durch den Einfluss verteilter Systeme und der Objektorientierung im Systementwicklungsprozess befindet sich die Wissenschaft Informatik in einer Phase des Paradigmenwechsels. Sie hat im Laufe ihrer Entwicklung eine fachliche Identität entwickelt und konturiert zunehmend ihr Selbstbild. Es folgt eine Darstellung der Konsequenzen dieser Entwicklung für das Schulfach Informatik. Der Paradigmenwechsel findet Eingang im Informatikunterricht, indem er in konkrete Lehr-/Lernprozesse umgesetzt wird. Ausgewiesene Zugänge werden mit konkreten Modellierungen verbunden und exemplarisch in ihrer unterrichtlichen Realisierung vorgestellt. Auf fachdidaktischem Hintergrund ergeben sich Konsequenzen für curriculare Elemente eines zweistündigen Pflichtfachs Informatik in der Sekundarstufe I (Jahrgänge 5 bis 10). Die unter den aktuellen Bedingungen mögliche Ausgestaltung des Unterrichts in der Sekundarstufe II wird mit dem Fokus auf ein spiralisches Curriculum – orientiert an Leitlinien dokumentiert.

**Abstract** During its development computer science has more and more established a way it sees itself. However, in recent years the rise of distributed systems and object orientation has initiated a paradigm change of computer science. This paper discusses the effects of this development on the computer science curriculum in German schools. The paradigm change has an effect on computer science lessons by influencing concrete teaching and learning processes. We combine specific teaching approaches with concrete models and present their realisations in school by several examples. From the didactic perspective these results give hints for the curricular elements of a two-hour compulsory course of computer science for secondary schools (grades 5 to 10 resp. age 10 to 15) in Germany. Based on these considerations and the situation in Germany we also propose a spiral curriculum for computer science for grades 11 to 13 in secondary schools.

## Vorbemerkungen<sup>1</sup>

Bei diesem Beitrag handelt es sich um die überarbeitete Fassung von [Humbert 1999a]. Er wurde u. a. um Quellenangaben erweitert, die im Internet verfügbare weitergehende Informationen zu ausgewählten Teilbereichen anbieten.

25 Jahre nach Einführung des Fachs Informatik in der gymnasialen Oberstufe ist es an der Zeit, eine Bestandsaufnahme zu leisten, die die Perspektiven dieses Fachs aufzeigt. Bis heute kann die Hochschulinformatik, aber auch die Berufsbildung, nicht auf solides und fundiertes Vorwissen im Bereich der Informatik aus der Schule zurückgreifen.

---

<sup>1</sup>Im Text wird - abgesehen von Zitaten - durchgängig das *generische Femininum* verwendet. Männer mögen sich dadurch nicht ausgeschlossen fühlen.

Um dieser für den Wirtschaft- und Produktionsstandort Deutschland kontraproduktiven – weil z. B. studienzeitverlängernden – Situation zu begegnen, ist eine Abstimmung der Inhalte und Methoden eines verpflichtenden informatischen Curriculums für die Sekundarstufen I und II unabdingbar.

In allen Bereichen moderner Industrie- und zunehmend **Wissensgesellschaften** ist grundlegendes informatisches Strukturwissen inzwischen unverzichtbar für erfolgreiches Lernen und Arbeiten.

Dem vorliegenden Beitrag geht schulpraktische Arbeit des Autors voraus. Den in **2 Umsetzung . . .** dargestellten Elemente des konkreten Unterrichts in der gymnasialen Oberstufe liegt konkreter Informatikunterricht zu Grunde.

## 1 Geschichte

### 1.1 Informationelle Selbstbestimmung auf der Grundlage der Allgemeinbildung

An dieser Stelle soll nicht die Diskussion um den allgemeinbildenden Charakter der informatischen Bildung zusammengefasst werden. Es geht hier vielmehr darum, exemplarisch und plakativ konkrete Anforderungen an die informatische Bildung deutlich zu machen.

Informatik ist in den entwickelten Gesellschaften zunehmend eine konstitutive Grundlage für zielgerichtetes Handeln auf verschiedenen Ebenen. Dies ist keineswegs eine neue Erkenntnis: siehe z. B. [Nora und Minc 1979].

Dies spiegelt sich in der Veränderung der Anforderungen an die Schule über Fachgrenzen hinweg wider. Eine demokratisch verfasste Gesellschaft muss allen Bürgerinnen die Voraussetzungen für die Teilnahme an wichtigen Entscheidungsprozessen bieten.

Gesellschaftliche Fragestellungen, die ohne grundlegendes Verständnis informatischer Zusammenhänge nur unzureichend verstanden und damit diskutiert werden können, sind beispielsweise das Urteil des Bundesverfassungsgerichts zum Recht auf informationelle Selbstbestimmung und die Krypto-Debatte.

Wieviel Wissen über Informatik braucht die „mündige Bürgerin“, um ihre Rechte wahrnehmen zu können und um sachkundig an zentralen Diskussionen über die Gestaltung der Zukunft mitzuwirken?

### 1.2 Entwicklung der Informatik an Hochschulen

#### Herausbildung der Informatik als Wissenschaftsdisziplin in Deutschland

Bei der Analyse der inzwischen „verschütteten“ Informatik-Diskussion wird deutlich, dass sich auf dem Hintergrund historischer Betrachtungen der Begründungen für die Etablierung der Informatik einige Elemente des Kerns der Informatik wesentlich deutlicher herauschälen lassen als durch die Betrachtung des Status-Quo (vgl. [Krabbel und Kuhlmann 1994]).

Zunächst werden die Teile dargestellt, die das heutige Bild der Informatik prägen:

#### Fundamentale Grundbegriffe der Informatik<sup>2</sup>

BRAUER weist daraufhin, dass viele informatische Fragestellungen in der Geistesgeschichte eine lange Tradition haben: „Fundamentale Grundbegriffe der Informatik, wie der Begriff des mechanisch ablaufenden Prozesses, der Begriff des Algorithmus, die Ideen der formalen Beschreibung (etwa der Regeln des logischen Schliessens) und der Konstruktion künstlicher Sprachen, sowie der Mechanisierung angeblich geistiger Tätigkeiten (wie das Rechnen mit Zahlen oder das Umformen algebraischer Ausdrücke) und die Versuche, Automaten zu bauen, gehen zum Teil bis auf das griechische Altertum zurück ...“ [Brauer u. a. 1980, S. 44].

In der gleichen Veröffentlichung wird eines der Grundprobleme der Informatikausbildung deutlich formuliert, nämlich: „daß wir nicht über Verfahren zur systematischen Erstellung von zuverlässigen Programmsystemen verfügen und daß in der Praxis der Datenverarbeitung die Anwendung empirischer Regeln die Verwendung wissenschaftlich fundierter Methoden weit übertrifft.“ ([Brauer u. a. 1980, S. 45], aber auch [Brauer u. a. 1989, S. 55f])

---

<sup>2</sup>nicht zu verwechseln mit den **Fundamentalen Ideen der Informatik**, wie sie SCHWILL beschreibt, siehe 1.3

## Teilgebiete der Informatik

Von Volker CLAUS wurde in der Mitte der 70er Jahre ein Begriffsskelett für die verschiedenen Teilgebiete der Informatik vorgestellt und begründet (siehe [Claus 1975, S. 11]). Er ordnet die verschiedenen Teilgebiete der Informatik den beiden Kategorien **Kerninformatik** und **Angewandte Informatik** zu. Die Kerninformatik wird weiter in die drei Untergruppen **Theoretische**, **Technische** und **Praktische** Informatik aufgespalten. Damit stand eine, wenn auch pragmatisch orientierte, inhaltliche Struktur zur Verfügung. Diese Einteilung wurde Ende 1999 vom FAKULTÄTENTAG INFORMATIK zu Gunsten der Unterteilung in „Grundlagen der Informatik“, „Informatik der Systeme“, „Angewandte Informatik“ und „Zusatzkompetenzen“ geändert. „Dabei wurde von der klassischen Einteilung [...] abgewichen, weil sich die Unterschiede immer mehr verwischen. Modellierungstechniken machen zum Beispiel keinen Unterschied zwischen Soft- oder Hardware.“ [Fakultätentag Informatik 1999]

## Konzepte, Methoden und Paradigmen der Informatik

### Entwurfsmodelle:

Die Konstruktion von Softwaresystemen<sup>3</sup> ist eine Ingenieuraufgabe. Eine Phasierung dieser Aufgabe findet ihren Ausdruck in dem sogenannten Wasserfallmodell:

[Problem-]Analyse, Entwurf, Implementierung [und Testen], Integration, Installation und Wartung [Kroha 1997, S. 28]

Wie sich im Laufe der Entwicklung herausgestellt hat, muss der mit dem Wasserfallmodell verbundene streng hierarchische Weg an einigen Stellen aufgebrochen werden - z. B. um Entwurfsentscheidungen frühzeitig erkennen zu können (Rapid Prototyping) oder um den Entwicklungsprozess durch Rückkopplung auch in späteren Phasen beeinflussbar zu halten. [Appelrath u. a. 1998, S. 110]

**Programmiersprachenparadigmen und ihre Ausprägung in Sprachklassen:** Für große Teile der Informatikausbildung sind die Phasen **Entwurf** [-spezifikation] und **Implementierung** von zentraler Bedeutung. Jede der Zuordnungen, die in Tabelle 1 dargestellt sind, impliziert eine spezielle Methode<sup>4</sup> der informatischen Modellierung. Eine Übersicht für mögliche Umsetzungen im Schulfach Informatik findet sich in [Bosler 1992, S. 157-224].

Tabelle 1: Auffassungen und ihre Ausprägung in Sprachklassen, nach [Padawitz 1995, S. 5], [Padawitz 1998]

Auffassung	Sprachklasse <sup>5</sup>
Auswertung von Ausdrücken (einer formalen Sprache)	funktionale und applikative Sprachen <sup>6</sup>
Beantwortung von Anfragen (an ein Informationssystem)	relationale und logische Sprachen <sup>7</sup>
Manipulation von Objekten (der realen Welt)	prozedurale, imperative und objektorientierte Sprachen <sup>8</sup>

<sup>3</sup>Der Begriff **System** hat inzwischen in einer Form Eingang in die Normung gefunden, dass Softwaresysteme eingeschlossen sind (vgl. DIN 19226).

<sup>4</sup>Um Begriffsverwirrungen vorzubeugen: hier sind keine pädagogischen Methoden im Blick.

<sup>5</sup>Kurzcharakterisierung der Sprachklassen und Hinweise auf Umsetzungen für das Schulfach Informatik finden sich in den folgenden Fußnoten:

<sup>6</sup> Alle Objekte werden als Funktion  $f : D \rightarrow E$  aufgefasst, selbst Konstante sind konstante Funktionen. Funktionen können Argument einer weiteren Funktion sein und vice versa. Siehe z. B. [Staudte 1992, Loethe 1997].

<sup>7</sup> Mit Hilfe des Prädikatenkalküls wird versucht, über Aussageformen Mengen zu beschreiben. Damit kann der Zusammenhang zwischen der mengentheoretischen und der aussagenlogischen Beschreibung von Problemstellungen zum Zweck der Bestimmung von Lösungsmengen geleistet werden. Siehe z. B. [Baumann 1991, Schubert 1992].

<sup>8</sup> **imperativ:** Im Mittelpunkt steht die an virtuellen Maschinen orientierte Programmiersprache; Ziel ist das Erlernen einer speziellen Programmiersprache - die Problemstellungen werden dem einzuführenden „Befehl“ angepasst. Dieser Ansatz wurde im Laufe der Zeit zunehmend in Richtung auf Strukturen hin verbessert und optimiert und stellt heute an vielen Schulen im Sekundarstufen II-Bereich den abschließlichen Standard des Informatikunterrichts dar. Siehe z. B. [Balzert 1983]

**objektorientiert:** Jedes System besteht aus Objekten. Jede Aktivität des Systems ist die Aktivität einzelner Objekte. Gleichartige Objekte werden zu Klassen zusammengefasst. Die Objekte stellen eine Einheit aus *Zustandsvariablen* und *Methoden* dar, d.h. im Objekt wird die imperative Trennung von Daten und Aktion auf (mit) den Daten [bei der Analyse] aufgehoben. Zentral ist die Analyse des Gegenstandes und der anschließende Entwurf, allerdings kommen zentrale imperative Konzepte bei der Erarbeitung von Methoden durchaus zum Einsatz. Siehe z. B. [Müller 1992, Czischke u. a. 1999]

**Sichtweisen der Informatik in der Diskussion** (umfassend dargestellt in: [Coy u. a. 1992])

Die jüngeren Auseinandersetzungen um den curricularen Kern der universitären Informatikausbildung zwischen PARNAS [Parnas 1990], DIJKSTRA [Dijkstra 1989] und anderen sind von COY zusammenfassend und pointiert dargestellt worden (siehe [Coy 1992, S. 3f]). Die Ursachen für die unterschiedlichen Sichtweisen liegen sowohl auf der begrifflichen Ebene (Fach-Begriffe, ihre Definitionen, ihre fachliche Konkretion) wie auch auf der fachpraktischen Ebene (welches an *Produkten* interessierte Fach kann es sich auf Dauer leisten, ihre „Fach“arbeiterinnen so schlecht auszubilden, dass immer noch Programme abstürzen, ... [Coy 1992, S. 4]). In der Diskussion um eine neue Theorie der Informatik<sup>9</sup> lassen sich folgende Argumentationsstränge ausmachen: die Wirkungen von Informatiksystemen und die Verantwortung der Informatikerin; der disziplinäre Kern der Informatik als technischer Umgang mit Wissen bzw. Informationen bzw. Sprache zur Codierung von Vorstellungen oder der Eigenschaft des Computers als symbolverarbeitende Maschine und nicht zuletzt der Sinn und Zweck der Informatik [Rolf 1992, S. 33]. Eine ausführliche Darstellung findet sich in der bereits zu Beginn dieses Abschnitts erwähnten Diplomarbeit: [Krabbel und Kuhlmann 1994]. Dort werden von den Autorinnen die in der aktuellen Diskussion der Informatik zentralen Begriffe: **Wissen, Information, System/Realität, Sprache** und **Gestaltung** herausgearbeitet.

### 1.3 Informatik in der Schule

Die Dynamik der Neuorientierung der Informatik spiegelt sich auch in der Schule wider. Der überwiegende Anteil der Kolleginnen, die das Schulfach Informatik vertreten [müssen], sind jedoch keine grundständig ausgebildeten Informatiklehrerinnen. Deshalb steht zu befürchten, dass die folgenden Anmerkungen noch einige Zeit gültig bleiben werden.

Anmerkung: Grundständig ausgebildet bedeutet, dass die Informatiklehrerinnen sowohl das erste, wie auch das zweite Staatsexamen in Informatik erworben haben. Fortbildungen, die teilweise auf fachlich hohem Niveau durchgeführt wurden, ersetzen nicht die notwendigen fachdidaktischen Praxisanteile der ersten und zweiten Phase der Lehrerbildung, die gerade für das Schulfach Informatik unabdingbar sind. Die Primärqualifikation von Informatiklehrerinnen muss quantitativ massiv ausgeweitet werden.

Einige Zahlen zur aktuellen Situation der Lehrerbildung wurden in [Humbert 2000a] dokumentiert.

- Informatikprofessorinnen charakterisieren die Ergebnisse der Informatik in der Schule folgendermaßen: „die Schulinformatik ist derzeit eher kontraproduktiv. Sie vermittelt ein Bild von der Informatik, das den Schülern eine falsche Basis für ihre Studien-Entscheidung an die Hand gibt. [...] Die Schüler brächten zum Teil ein hervorragendes Spezialwissen mit; es sei aber für eine große Schülerzahl nicht möglich, die Brücke von den Formalkenntnissen zur eigenen Lösungsfähigkeit zu schlagen (ein Problem sicherlich nicht nur der Informatik)“ [König 1993, S. 6].
- „Für die Informatik stellt sich dabei die Frage: Wie hilft man beim Einstieg in dieses Fach, über das – etwa im Gegensatz zu Mathematik und Physik – bei Schülerinnen und Schülern oftmals keine oder falsche Vorstellungen herrschen?“ [Appelrath u. a. 1998, Klappentext]

Ein differenzierteres und empirisch abgesichertes „Bild der Informatik“ von Informatiklehrerinnen vermittelt [Berger 1997].

#### Zugänge zur Informatik

Ein Spiegel der Bedeutung der Informatik und ihres Selbstverständnisses ist - mit einer gewissen Verzögerung - in der Ausgestaltung der Lehre zu finden. Dies betrifft sowohl die universitäre wie auch die schulische Informatik. Dabei haben Einflüsse der universitären Informatik das Schulfach Informatik vorangetrieben. Allerdings ist dieser Einfluss über die Zeit stark zurückgegangen. In der Tabelle 2 werden die gängigen Zugänge (oft auch Orientierungen genannt) zur Erarbeitung informatischer Inhalte aufgelistet.

---

<sup>9</sup>nicht zu verwechseln mit der oben erwähnten theoretischen Informatik (2 Teilgebiete der Informatik)

Tabelle 2: Zugänge zur Erarbeitung informatischer Inhalte

Zugang <sup>10</sup>	Ort	Literatur	Zeitpunkt
Hardware <sup>11</sup>		[von Cube 1960, Meißner 1975]	1960
Algorithmen <sup>12</sup>	Gelsenkirchen	[CUU-Gruppe Gelsenkirchen 1973]	1973
		[Gesellschaft für Informatik e. V. 1976]	1976
Anwendung <sup>13</sup>	Berlin	[Arlt und Koerber 1980]	1979
Gesellschaft <sup>14</sup>		[AG Neue Medien in der GEW NRW 1989]	1988
Software-Modifizierung <sup>15</sup>	Berlin	[Lehmann 1993]	1993
Information <sup>16</sup>	München	[Hubwieser und Broy 1997, Baumann 1996]	1996

An dieser Stelle soll auf die konzeptionelle Arbeit zu den **Fundamentalen Ideen der Informatik** [Schwill 1993] hingewiesen werden, der eine quer zu diesen Zugängen liegende Struktur benennt und für die Didaktik der Informatik nutzbar macht.

Bei diesem Vorschlag für die Fundamentalen Ideen bleibt allerdings der Diskussionsprozess um die Ränder, die Abgrenzung, das Selbstverständnis und die Sichtweisen der Informatik außen vor.

### Programmiersprachen und der Einsatz von Tools in der [Schul-] Informatik

Ohne die Möglichkeit der Umsetzung von erarbeiteten Problemlösungen, also Realisierung der entwickelten „abstrakten Maschine“ kann die Informatik in der Schule nicht bestehen. Deshalb spielen die [software-] technischen Hilfsmittel eine große Rolle für den praktischen Unterricht. Für alle **Sprachklassen** existieren Hinweise für die Umsetzung im Informatikunterricht (siehe die Anmerkungen zur Tabelle 1).

In der Entwicklung der Informatik in der Schule können verschiedene Sprach-Phasen unterschieden werden:

- die *Frühphase*: Arbeit mit BASIC - 70er Jahre,
- die *Phase des Sprachenstreits*: Beginn der Arbeit mit PASCAL in verschiedensten Ausprägungen, Verdrängung von BASIC; die Arbeit orientiert sich an „Go To Statement Considered Harmful“ [Dijkstra 1968],
- die *Phase der Ausdifferenzierung*: ELAN, UCSD-Pascal, Turbo Pascal, Think Pascal, ...,
- Modularisierung: Modula-2 - das *Konzept der Abstrakten Datentypen* wird nach und nach in den Informatikunterricht integriert - 80er Jahre,
- *Objektorientierung* in der Schule wird programmtechnisch möglich: Object Pascal, Oberon, Turbo-Pascal ab Version 5.5, *Konzepte zur Objektorientierung* im Anfangsunterricht (z.B. Stifte und Mäuse [Czischke u. a. 1999]) werden entwickelt – 90er Jahre, in der Schule finden zunehmend „*Visuelle Programmierumgebungen*“, wie Visual Oberon, Visual Basic oder Delphi Verbreitung, allerdings i. d. R. ohne eine konzeptionelle Grundlage für die unterrichtliche Umsetzung.

<sup>10</sup> der Zugang „Problem“ ist nicht gesondert in der Tabelle aufgeführt - nach der Darstellung von [Baumann 1996, S. 220]: Behandlung von Problemen, die den Prozess der Informatisierung bereits „hinter sich“ haben, bzw. keiner Informatisierung mehr bedürfen, da sie im inner-informatischen Kontext entstanden sind – diese Definition ist allerdings in dieser eingeschränkten Form mit allgemeinen Didaktikansätzen unverträglich: danach kann jeder der hier angegebenen Zugänge problemorientiert gestaltet werden.

<sup>11</sup> grundlegende Rechnerstrukturen; eine der Kybernetik verpflichtete „Rechnerkunde“

<sup>12</sup> Nachdem die ersten Vorüberlegungen der GI [Gesellschaft für Informatik e. V. 1976] bekannt wurden, fand ein Fokussierung der Entwicklung der Lehrplan- und Richtlinienentwicklung für die Sekundarstufe II auf den algorithmenorientierten Ansatz statt. Allerdings blieb der berufsbildende Bereich von dieser Entwicklung lange ausgenommen: Erst 1983 [Fachausschuß „Ausbildung“ der Gesellschaft für Informatik 1983] wird eine „Sofortmaßnahme“ für die „Einbeziehung von Informatikinhalten in die berufliche Erstausbildung an gewerblich-technischen berufsbildenden Schulen“ gefordert, die in einer Unterrichtsstunde pro Woche realisiert werden soll. „Dabei ist das Wissen über den detaillierten Schaltungsaufbau der Mikroelektronik unwesentlich.“

<sup>13</sup> Orientierung auf informatische Modellbildung, Übertragung von realen Prozessen auf die informatische Ebene und Rückbezug der informatischen Prozesse auf die Ausgangssituation.

<sup>14</sup> Zielt auf Analyse der Auswirkungen der Informatik auf die Gesellschaft, indem für gesellschaftlich zentrale informatische Fragen und Entwicklungen eine fachliche Grundlage erarbeitet wird.

<sup>15</sup> Modifizierung: Ein komplexes, existierendes System wird gewartet und erweitert.

<sup>16</sup> Informationen müssen formal repräsentiert, verändert und transportiert werden, um anschließend wieder interpretiert zu werden. (vgl. [Hubwieser 1998, S. 7])

Diese sind für die konkrete, praktische Arbeit der Kolleginnen in den Schulen deshalb bedeutsam, als sich Veränderungen in diesem Bereich direkt auf die konkrete Unterrichtsvorbereitung und -durchführung und nicht zuletzt auf die Wiederverwendung erstellter Materialien auswirkt (vgl. z. B. [Füller 1999]).

## 2 Umsetzung: Informatik in der Schule und für die Schülerinnen

Die im Unterricht eingesetzte Programmiersprache kann die didaktische Umsetzung von Grundkonzepten erleichtern oder erschweren. Sie kann aber kein Konzept ersetzen. Im Folgenden werden Beispiele aus der unterrichtlichen Praxis dargestellt, bei denen bereits im Planungszusammenhang das Augenmerk auf bestimmte Zugänge gerichtet wird. In der Tabelle 3 werden die dokumentierten Unterrichtssequenzen charakterisiert. Dabei sind die Grenzen zwischen der *Gesellschaftsorientierung* und der *Anwendungsorientierung* nicht so trennscharf, dass der gewählte Zugang immer eindeutig zuzuordnen ist. In allen Beispielen kommt der Modellierung des jeweiligen Zusammenhangs eine zentrale Bedeutung zu.

Tabelle 3:Beispielsequenzen<sup>17</sup>

Zugang	Modellierungsmethode	Sprache	Beispiel
Gesellschaftsorientierung	objektorientiert	Python	Kooperative Arbeit
Anwendungsorientierung	objektorientiert	Python	Analyse und Auswertung von Daten
Anwendungsorientierung	logisch	Prolog	Modellierung eines Labyrinths

Die jeweils nötigen System- und/oder Sprachkenntnisse wurden integrativ erarbeitet, damit nicht auf Vorrat gelernt werden muss. Bezüglich der pädagogischen Prozesse fühle ich mich der pragmatisch-konstruktivistischen Tradition verbunden (siehe z. B. [Duit 1995]).

Seit der Diskussion um TIMSS [Baumert u. a. 1998] und den für die Unterrichtspraxis relevanten Erkenntnissen (abgeleitet aus der mangelhaften Problemlösekompetenz der bundesrepublikanischen Schülerinnen) gewinnt die konstruktivistische Sichtweise aus der naturwissenschaftsdidaktischen Forschung zunehmend den ihr zukommen Stellenwert in der pädagogischen Praxis des naturwissenschaftlichen Unterrichts. Für die Unterrichtspraxis bedeutet das u. a.: die Erarbeitung von Problemlösungen erfolgt nicht in Einzelarbeit, die Vorerfahrungen der Schülerinnen werden thematisiert.

Weitere Anregungen vor allem bzgl. der „Kognitiven Aspekte Sozialen Lernens“ finden sich in [Hollenstein 1998].

### 2.1 Kooperative Arbeit – CSCW-Systeme in der Schule

Gruppenarbeitsverfahren dienen u. a. der gemeinsamen Nutzung von verteilten Dokumenten. Im Zusammenhang mit zunehmender Vernetzung und Vereinheitlichung der Endbenutzungsoberflächen (Browser) können sie auch in der Schule genutzt werden. Grundlegende Ideen für eine unterrichtliche Umsetzung:

Motivation durch die **Schreibtischmetapher**: Virtuelle Gruppenarbeit lässt sich durch einen gemeinsamen Schreibtisch symbolisieren, auf dem Dokumente und Ordner liegen. Das CSCW-System enthält eine *Dokumentenverwaltung*, mit der die *Gemeinsame Dokumentenbearbeitung*, die *Kommunikation der Gruppenmitglieder* und der Prozess des *Aufmerksammachens (Awareness)* unterstützt werden.

Der Unterricht für die Einführung in die Arbeit mit einem CSCW-System ist in [Humbert 1999b] dokumentiert.

In künftigen Arbeitszusammenhängen wird die Nutzung dieser technischen Unterstützung nicht mehr auf wenige beschränkt bleiben. Durch Internetkonnektivität wird virtuelle Gruppenarbeit zu einem mächtigen Hilfsmittel bei der Organisation gemeinsamer Arbeit an verschiedenen Orten und mit unterschiedlichen Hard- und Softwaresystemen.

Durch die zunehmende Verfügbarkeit schulischer Intranets kommt der Informatik in der Schule die Aufgabe zu, technisch sinnvolle Möglichkeiten nutzbringend in einer schulischen Umgebung verfügbar zu machen und exemplarisch einzusetzen.

<sup>17</sup>Um Redundanzen zu reduzieren, wird in diesem Beitrag auf die Darstellung des gesellschaftsorientierten, imperativen, in Modula-2 umgesetzten Beispiels Automatentheorie verzichtet (siehe [Humbert 1999a, S. 179ff]).

Die im hier dokumentierten Unterrichtszusammenhang genutzte Infrastruktur wird in [Humbert 1998c] dargestellt. Weitergehende Überlegungen und notwendige Anforderungen an ein schulisches Intranet werden in [Humbert 2000b] vorgestellt.

Der Zugang erfolgt gesellschaftsorientiert.

Die gesellschaftsorientierte Variante setzt die Analyse von Aufbau- und Ablauforganisation in dokumentenorientierten Arbeitsabläufen voraus. Damit lassen sich Perspektiven für zukünftige Änderungen in diesen Bereichen handelnd erschließen und fundiert prognostizieren.

Die weiteren Schritte in der **informatischen** Arbeit in der Schule sind von der Erweiterbarkeit des zugrundeliegenden CSCW-Systems abhängig.

In diesem Kurs wurde das BSCW-System der Gesellschaft für Mathematik und Datenverarbeitung mbH eingesetzt: <http://bscw.gmd.de/>. Das BSCW-System ist inzwischen in den ODS-Kommunikationsserver (kostenfreie Schulserverlösung) <http://www.heise.de/ct/schan/> integriert worden.

In der konkreten Ausprägung sind CSCW-Systeme i. d. R. objektorientiert und modular strukturiert, so dass ein Einsatz im Informatikunterricht möglich ist, bei dem ein solches System experimentell anwendungsorientiert erweitert wird.

Im Unterschied zur Frühzeit der Informatik in der Schule sollte vermieden werden, Erweiterungen an solchen Systemen mit überzogenen Anforderungen zu überfrachten. Die Zeiten, in denen im Informatikunterricht Produkte erstellt wurden, die Ausgangspunkt für eine außerschulische Nutzung darstellten, sind durch die Komplexität heutiger Systeme vorbei, auch wenn immer wieder Schülerinnen ausgehend von ihrem konkreten Informatikunterricht Software mit hervorragender Funktionalität entwickeln. So zeigt eine grobe Übersicht der bei *Jugend forscht* im Segment Mathematik/Informatik eingereichten Beiträge, dass diese fast ausschließlich aus dem Bereich Informatik stammen.

Zusammenfassend läßt sich festhalten, dass CSCW-Systeme in den mir bisher vorliegenden konkreten Unterrichtserfahrungen eher vom Werkzeugcharakter eine nicht zu unterschätzende Rolle spielen sollten [Humbert 1998a].

Der Einsatzbereich *Ausbildung von [Informatik-] Lehrerinnen* wird in [Humbert 1998b] <http://in.Hagen.de/humbert/vortraege/seminar/welcome.html> beschrieben. Inzwischen wurde das Rahmenkonzept „Neue Medien in der Lehrerausbildung“ vorgelegt, in dem ausdrücklich die Notwendigkeit kollaborativer Elemente in der Infrastruktur für eine zukunftsweisende Lehrerinnenausbildung gefordert wird (vgl. [MSWWF 2000, S. 69f]).

Da ich die Erweiterung eines CSCW-Systems bisher unterrichtlich nicht umsetzen konnte, möchte ich anregen, auf diesem Feld unterrichtliche Erfahrungen zu sammeln.

Dazu ist es notwendig, dass die Klassen in einer Weise dokumentiert sind, die es der interessierten Lehrerin gestattet, im Unterrichtskontext mit den Schülerinnen Erweiterungen vorzunehmen, auf ihre Funktionalität zu untersuchen und sie ggf. der Schulgemeinde im schulischen Intranet zur Verfügung zu stellen. Dies führt zu Überlegungen, die die Verfügung über dokumentierten Quellcode der in der Schule eingesetzten Software als Voraussetzung eines gestaltenden Informatikunterrichts deutlich werden lassen.

## 2.2 Problemstellung: Analyse und Auswertung von Daten

In der Nähe der Gesamtschule Haspe, Hagen befindet sich eine Turbine, die von dem Verein für Regenerative Energien betrieben wird. Die Turbine produziert neben elektrischem Strom auch einen Strom von Daten. Diese Daten sollen in eine andere Form konvertiert werden, damit sie z. B. im Internet verfügbar gemacht werden können und darüber Auskunft geben, wieviel Strom aktuell produziert wird, wie sich die Wassergeschwindigkeit entwickelt, etc. Im Turbinenhaus steht andere Hard- und Software zur Verfügung als in der Schule. Als Hilfsmittel für die Umwandlung kam aufgrund der Rahmenbedingungen eine Skriptsprache in Betracht.

Ich wählte die objektorientierte, interpretierte Skriptsprache Python [Löwis und Fischbeck 1997, Lutz und Ascher 2000], die es erlaubt, betriebssystemnahe Operationen, wie das Lesen und Schreiben von Dateien betriebssystemunabhängig zu formulieren, so dass die Programmentwicklung sowohl in der Schule (MacOS) wie auch an der Turbine (DOS) und für den Linux-Server der Schule möglich wurde.

### Unterrichtsgang

Dauer der Sequenz: 10 Wochen in einem Grundkurs Informatik 11.2 - mit Vorbesprechungen, Begehungen und Schülerfachvorträgen.

Auf Grund technischer Probleme wurden die angekündigten Daten nicht geliefert. Daraufhin entschieden die Schülerinnen, die Projektidee (Analyse und Auswertung von Daten) an anderen Daten umzusetzen:

Als Idee formulierte eine Schülerin: Nehmen wir doch die Daten aller Hagerer Schulen.

Damit wurde die Aufgabe formuliert, Schuldaten - vermittelt über eine Klassenstruktur – so abzubilden, dass am Ende automatisch strukturierte und formatierte Ergebnisse produziert werden. Dies entspricht in Teilen der ursprünglichen Idee und nutzt nicht nur der eigenen Schule, sondern allen, die ein Verzeichnis der Hagerer Schulen benötigen.

Sequenzierung der Reihe:

Über reduzierte Problemstellungen wurden zentrale Elemente der objektorientierten Analyse erarbeitet und an konkreten Beispielen vertieft. Mit Hilfe von Schülerfachvorträgen erarbeiteten die Schülerinnen ausgewählte Elemente von HTML, die für diesen Zusammenhang notwendig erschienen. Integriert in den Unterrichtsablauf wurden Entwicklung und Implementierung der folgenden Klassen durch die Schülerinnen geleistet.

- Klasse *Datenstruktur*, deren Objekte die logische Datenstruktur eines Datensatzes abbilden und textuell ausgeben:
- Klasse *HtmlSeite*, deren Objekte eine HTML-Seite erzeugen:

<b>Datenstruktur</b>
<pre> __init__(art, name, strasse, tel, url, email) setArt (neueArt) setName(neuerName) setStrasse(neueStrasse) setTel(neueTel) setUrl(neuerURL) setEmail(neueEmail) zeigeAn()</pre>

<b>HtmlSeite</b>
<pre> __init__() erzeugeHeader(bgcolor="") erzeugeBody() # abstrakt erzeugeEnde() erzeuge(bgcolor="")</pre>

- Klasse *DatenstrukturListe*, deren Objekte aus einer Textdatei eine Liste (Datenstruktur) erzeugen und füllen; weitere Funktionen: die Liste nach verschiedenen Kriterien sortieren, erzeugte Liste ausgeben:
- Klasse *Seite*, die eine Erweiterung der Klasse *HtmlSeite* ist und deren Objekte eine HTML-Seite generieren und in einer Datei ablegen (dazu muss die leere - abstrakte – Methode *erzeugeBody* aus *HtmlSeite* überschrieben und damit gefüllt werden):

<b>DatenstrukturListe</b>
inhalt[]
<pre> __init__() erzeugeAusDatei (file) sortiereNachArt() sortiereNachStrasse() sortiereNachName() zeigeAn()</pre>

<b>Seite (HtmlSeite)</b>
<pre> __init__() setSchulliste(DatenstrukturListe) erzeugeBody() erzeugeEnde()</pre>

Da die Klassen jeweils auch einen Abschnitt zum Selbsttest umfassen, liessen sich im Laufe der Erarbeitung die Phasen der Modellierung, der Implementierung und des Tests gut miteinander verzahnen.

Ergebnisse, die regelmässig mit Hilfe dieser Schülerarbeit erzeugt werden, lassen sich auf den Seiten `schulen_nach_art.html`, `schulen_nach_namen.html` und `schulen_nach_strassen.html` unter <http://www.ha.nw.schule.de/schulen.html> finden.



## 2.3 Modellierung mit dem logischen Ansatz im Anfangsunterricht für alle Schülerinnen

Die Schülerinnen entwickeln - ausgehend von der nebenstehenden Abbildung - Fragestellungen, analysieren die Problemlage und setzen diese mittels logischer Modellierung um.

Von den Schülerinnen formulierte Fragestellungen:

Welches ist der kürzeste Weg?

Welches ist der schnellste Weg?

Wie können Sackgassen erkannt werden?

Wie kommt die Figur durch das Labyrinth?

Zur Umsetzung muss der Lösungsraum bestimmt werden. Die Fakten - die Beschreibung der Teilstrecken des Labyrinths - lassen sich vereinfacht durch die Angabe von zwei (Anfangspunkt; Zielpunkt) zweistelligen (Zeilennummer Spaltennummer) Zahlen modellieren.

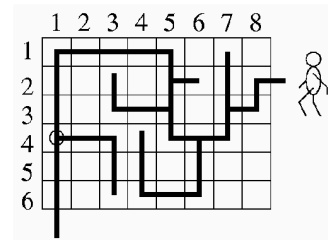


Abbildung 3: Das Labyrinth  
(siehe [Hans 1998])

Probleme bereiten den Schülerinnen die für die angemessene Strategie nötigen Constraints, die den Lösungsraum so einschränken, dass die Figur einen Weg durch das Labyrinth findet, indem eine Liste mit den bereits besuchten Punkten angelegt wird, die einen geordneten Rückzug ermöglicht.

Im Zusammenhang mit der ersten Lehrerausbildungsphase wurden in diesem Kurs die Schülerinnen sowohl mit dem Standardmodellierungsbeispiel (Familienbeziehungen – *Ahnentafel*) wie auch mit dem oben genannten Beispiel (*Labyrinth*) konfrontiert.<sup>18</sup> Die vorliegenden Erfahrungen berechtigen zu der Annahme, dass mit diesem Ansatz im Anfangsunterricht erfolgreich eine stark motivierende und zugleich punktuell durchaus in die Tiefe gehende Einstiegsphase in das informatische Modellieren geleistet wird. Dies wird durch die Auswertung einer Befragung der Schülerinnen am Ende des ersten Schulhalbjahres bestätigt: **zwei von sechs** befragten Gruppen machten deutlich, dass ihnen die Beschäftigung mit dem Thema *Ahnentafel* „am wenigsten gefallen“ hat, während **drei von sechs** der Gruppen das Thema *Labyrinth* in die Liste der Inhalte, „die mir besonders gut gefallen“ haben, aufgenommen haben.

## 3 Entwicklungslinien und Perspektiven

### 3.1 Zumutungen für die Schülerinnen

Um ein umfassendes Bild der Informatik zu vermitteln, fehlen fundierte Grundlagen bei den Schülerinnen, die in der Sekundarstufe I gelegt werden müssen:

Beginnen wir mit einer Schülerin, die sich 1999 in der 11. Jahrgangsstufe befindet:

Im 7. Jahrgang sollten alle Schülerinnen in 60 Unterrichtsstunden eine sogenannte informationstechnische Grundbildung erhalten. Diese wird fachfremd, integrativ unterrichtet von Kolleginnen, die z. B. Deutsch als Fach haben und z. B. den Schülerinnen die Bedienung einer Textverarbeitung soweit erklären, dass diese nun mit Hilfe des Computers Texte schreiben können.

An dieser Stelle könnten Grundprinzipien und -methoden der Informatik Eingang in den Unterricht finden, die eine Grundlage für alle Schülerinnen darstellen. Hier sei vor allem die durchgängige sinnvolle Bezeichnung der Erklärungen zur Nutzung von graphischen Benutzungsoberflächen hervorgehoben. Diese gemeinsame Grundlage kann z. B. durch eine an die objektorientierte Analyse angelehnte Sprechweise entwickelt werden.

Im weiteren Verlauf der Biographie der Schülerin kommt der Zeitpunkt, an dem sie sich dazu entschliesst, im Differenzierungsbereich (9. und 10. Jahrgang) das Fach Informatik zu wählen.

<sup>18</sup>Im Vorlesungsskript zur Didaktik der Informatik I der Universität Dortmund (für das Lehramt für die Sekundarstufe II) [Schubert 2000] finden sich Anhänge, die für den hier betrachteten Zusammenhang von Bedeutung sind: **B.3. Beispiel für prädikative Modellierung**, Seite 119-127 (enthält Elemente der Unterrichtsvor- und Nachbereitung dieses Tagespraktikums), **G. Logische (deklarative) Programmierung**, Seite 156-167

Wählt sie nun in der Oberstufe wieder Informatik, so sitzt sie in der Regel im 11. Jahrgang in einem sehr großen und vor allem sehr heterogenen Informatikkurs, der oft nicht über die 11 hinaus weitergeführt wird.

Wie müssen unterrichtbare Konzepte gestaltet sein, die unter solchen - zugegebenermaßen schlechten - Bedingungen den gesellschaftlichen Anforderungen genügen? Die curriculare und konzeptionelle Arbeit ist an den Schülerinnen zu orientieren, die faktisch die kürzeste Zeit in Informatikkursen verbringen. Dort wird das gesellschaftliche Bild der Informatik geprägt.

**Die persönliche Erstbegegnung mit der Informatik in der Schule determiniert die weiteren Erfahrungen.**

### 3.2 Vorüberlegungen zu einem Gerüst für ein Konzept unter den aktuellen Bedingungen

Ich gehe davon aus, dass die oben skizzierte Gesamtsituation in den nächsten Jahren nicht grundlegend geändert werden wird.

Inzwischen (April 2000) wird berichtet, dass ausgehend vom nordrhein-westfälischen Kultusministerium eine Initiative in der KMK gestartet wird, um das Fach Informatik den anderen Fächern des mathematisch-naturwissenschaftlichen Aufgabenfeldes bezogen auf das Abitur (Abdeckung der Pflichtbindung) gleichzustellen. Ausserdem ist zu vermelden, dass Bayern sich anschickt, ab dem Jahr 2003 Informatik in der Sekundarstufe I in Gymnasien verpflichtend einzuführen.  
Quelle: [http://ddi.cs.uni-dortmund.de/ddi\\_bib/presse/](http://ddi.cs.uni-dortmund.de/ddi_bib/presse/)

Deshalb setze ich auf eine konzeptionelle Variante, die die 11. Jahrgangsstufe zum Zentrum der Überlegungen macht:

Im 11. Jahrgang - zu Beginn der Oberstufe - wählen viele Schülerinnen das Fach Informatik, weil „man das ja irgendwie braucht“. Die Grundkurse schmelzen nach 11 auf unter 30%. Der Grund liegt darin, dass die Schülerinnen zur Zeit mit Informatik keine Pflichtbindung bzgl. des Abiturs abdecken können (differenzierte Schülerinnenanzahlen finden sich in [Humbert 2000a]).

- Wie sieht dieses Bild aus?
- Was hat dies durch den Informatikunterricht vermittelte Bild mit Informatik zu tun?

Allerdings muss unter den aktuellen Rahmenbedingungen versucht werden, ein halbwegs konsistentes und valides Curriculum zu entwickeln. In diesem kann für ein Schuljahr ein verpflichtender breiter Zugang zu informatischen Fragestellungen realisiert werden. Elemente daraus werden in einem spiraligen Curriculum in den darauf folgenden Jahrgangsstufen vertiefend wieder aufgenommen.

Betrachtet man z. B. den aktuellen, seit dem Schuljahr 1999/2000 geltenden Lehrplan Informatik für die Gymnasiale Oberstufe für Nordrhein-Westfalen [MSWWF 1999], so kann festgestellt werden, dass den Fachkonferenzen (und damit den Lehrerinnen) ein breiter Strauss an Varianten des Informatikunterrichts zum Angebot (und damit zur Auswahl) gemacht wird. Leider konnte bei der inhaltlichen Ausgestaltung des Lehrplans (3.4 Sequenzbildung: [MSWWF 1999, 44-71]) nicht eingelöst werden, was in dem 2. Teil: Bereiche, Themen, Gegenstände [MSWWF 1999, 10-35] mit guten Vorsätzen begann. Bei verantwortlichem Umgehen mit diesem Lehrplan durch engagierte Lehrerinnen besteht aber die Chance, dass den Schülerinnen dadurch ein umfassendes Bild der Informatik bereits in einem Schuljahr (11. Jahrgang) vermittelt wird. Damit verschiedene Sichten auf die Informatik ermöglicht werden können, erscheint bereits für einen Informatikdurchgang in einem Kurs, der ein Schuljahr Informatik in der 11. Jahrgangsstufe umfasst, ein Paradigmenwechsel unverzichtbar. Im Rahmen eines spiraligen Curriculums findet die Wiederaufnahme des zu Beginn gewählten Paradigmas auf erweiterter und erweiternder, vertiefender Ebene im zweiten Kursjahr statt.

Wichtiger als das gewählte Paradigma erscheint mir allerdings die grundlegende Sichtweise auf die Wissenschaft und das dadurch provozierte, prägende Bild, das Schülerinnen von der Informatik mitnehmen.

Informatikunterricht, der bisher keine breite Fachbasis in der Schule hat, steht vor einer schwierigen Aufgabe: Anforderungen verschiedenster Art strömen auf die Lehrerinnen ein, die Informatik unterrichten. Die Schülerinnen haben eine ausgeprägte Erwartungshaltung, die i. d. R. enttäuscht wird: Sie verlangen nach Bedienungshilfen für Installation von Hard- und Software, Bedienungshilfen für spezielle Produkte. Kaum je wird eine Schülerin verlangt, dass sie endlich Bäume ausbalancieren lernt. Es ist notwendig, den Informatikunterricht den Erfordernissen der aktuellen Ausprägung informatischer Systeme insofern anzupassen, dass Schülerinnen Vorerfahrungen nutzen können. Diese sollten aber unbedingt in einen von konkreten Oberflächen abstrahierten Kontext eingebunden unterrichtet werden.

Der Informatikunterricht bietet Möglichkeiten, die andere Schulfächer nur mit Mühe in ihre Fachdidaktik integrieren können:

- Es liegen Erfahrungen mit Schülerfacharbeiten (bereits in der Sekundarstufe I) vor.
- Es gibt eine Tradition projektorientierter Ansätze im Informatikunterricht.
- Im Informatikunterricht gibt es immer dann fachübergreifende Ansätze, wenn der für eine Problemsituation zu erschließende außerinformatische Fachzusammenhang schulisch verankert ist.
- Die kooperative Arbeit, die soziales Lernen befördert, ist Grundbaustein der Didaktik der Informatik.
- Der praktische Einfluss des Informatikunterrichts auf die Gestaltung des Schullebens ist für viele Schulen inzwischen unverzichtbar, so dass zunehmend schulscharfe Stellen<sup>19</sup> für Informatiklehrerinnen ausgeschrieben werden.

Diese - auf der Habenseite stehenden - Möglichkeiten gilt es in die Diskussion um die Zukunft der Bildung als konstruktive Beiträge der Informatik einzubringen.

### 3.3 Curricularer Vorschlag

**„Es ist sicher zu spät, die Schlüsselqualifikationen für den gesellschaftsverträglichen Umgang mit Informatik erst im Stadium der Hochschulausbildung anzugehen. Hier sind Kompetenzen gefragt, die im Rahmen der Erziehung und Ausbildung eines Menschen so früh wie möglich vermittelt und immer wieder eingeübt werden müssen. Sie lassen sich nicht nebenher erwerben.“**

Die Vermittlung und alltägliche Einübung der Basiskompetenzen und der Schlüsselkompetenzen an Schulen stellt besondere Anforderungen an die Ausbildung und Veränderungsfähigkeit der Lehrer und des Lehrsystems.“ [Breutmann 2000, S. 11, Hervorhebung im Original]

[http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Inf\\_LLL.pdf](http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Inf_LLL.pdf)

Informatik sollte in der Primarstufe (Jahrgangsstufen 1 bis 4) nicht als eigenes Fach ausgewiesen werden. Die vorfachliche Thematisierung von Elementen der Informatik wird dort z. B. im Fach Sachkunde, aber auch in fachlichen und vor allem in überfachlichen Zusammenhängen vorgenommen. Mit den zunehmenden technischen Möglichkeiten in der Grundschule muß allerdings unbedingt dafür Sorge getragen werden, dass die dort vermittelten Handlungselemente auf einer fachlich soliden Basis der Unterrichtenden umgesetzt werden. Vorfachliche Elemente müssen in den konkreten Kontext der handelnden Erarbeitung eingebettet werden.

#### 3.3.1 Erwartungen an einen künftigen Pflichtbereich Informatik in der Sekundarstufe I, Jgst. 5-10

Die folgenden Vorschläge stellen eine Übersicht von Elementen eines verpflichtenden Fachs Informatik für alle Schülerinnen in der Sekundarstufe I für die Jahrgangsstufen 5 bis 10 dar. Dabei wird von der Verankerung der Informatikanteile in der Sekundarstufe I als eigenständiges, durchgängiges, zweistündiges Fach ausgegangen.

<sup>19</sup>Erlauben den Schulen ein fach- und personenbezogenes Einstellungsverfahren für ihren konkreten Bedarf.

Die Vermittlung ist in einem spiraligen Curriculum zu realisieren, d. h. die hier gewählte Reihenfolge stellt keine Sequenz dar, sondern beschreibt die wesentlichen Bereiche, die der schulischen Umsetzung bezogen auf die konkreten Lerngruppen bedürfen.

Um grundlegende Handlungskompetenzen bei allen Schülerinnen zu ermöglichen, ist es unabdingbar, einführende Unterrichtseinheiten zur Arbeit mit der schulinternen, informatischen Infrastruktur (z. B. Netz – Mail-Funktion) durchzuführen. Schülerinnen erhalten mit ihrem Schülerschein einen schulbezogenen Mailaccount, mit dem sie personenbezogen handelnd im geschützten, schulinternen Netz (Intranet) grundlegende Kompetenzen erwerben. Diese werden im Laufe der Sekundarstufe I schrittweise erweitert. Weitere Elemente (Kompetenzbausteine) lassen sich durch angeleitete, fachlich untersetzte Exploration hinzufügen. Damit wird der gestaltende Umgang mit technisch unterstützter Kommunikation im geschützten, schulischen Umfeld ermöglicht.

Der Umgang mit Information als neuer Kulturtechnik ist ein grundlegendes Element des Pflichtbereichs Informatik. Über die digitale Repräsentation von Information werden die in der Informationsgesellschaft unabdingbaren fachbezogenen Grundlagen – didaktisch reduziert – zum Unterrichtsgegenstand. Schülerinnen entwickeln auf dieser Basis ein kognitives Modell von Informatiksystemen. Dabei muss berücksichtigt werden, dass die alten Metaphern Werkzeug, Medium, Denkzeug der Erweiterung um die Verdeutlichung der Dimensionen **selbstgesteuerter Automat** und **Dialogsystem** bedürfen. Auf diese Weise sind die Besonderheiten interaktiver Informatiksysteme stärker in den Blick zu nehmen. Dies ist notwendig, damit ein tiefergehendes Verständnis für die Besonderheit interaktiver Informatiksysteme erreicht wird.

Mit der altersgemäßen Entwicklung objektorientierter Bezeichnungen und Beschreibungen wird der intuitive Zugang zum informatischen Problemlösen durch Analyse und Konstruktion mittels objektorientierter Modellierung eingeleitet. Dabei ist der konzeptionelle Charakter zu betonen, da nach dem Paradigmenwechsel in der Informatik die objektorientierte Modellierung als Konzept der Informatik etabliert werden konnte.

Die unterrichtliche Umsetzung des Unterrichtsgegenstands **Sprache** schafft eine theoriegeleitete Verbindung des Schulfachs Informatik für die Anwendung in verschiedenen Unterrichtsfächern. Damit wird eine theoretische Klammer für verschiedene Unterrichtsfächer geschaffen. Fachübergreifende Ansätze erhalten so eine Abstützung im Fach Informatik. [Claus 1991]

Die Beherrschung von Komplexität ist eine der zentralen Zieldimensionen der Informatik. In der Sekundarstufe I sollte der Umgang mit großen Datenbeständen und Informationsräumen bezogen auf die Reduktion ihrer Komplexität erlernt werden. Dies erfolgt durch anwendungsbezogene, zielgerichtete Reduktion. In der konkreten Umsetzung bietet sich eine Auswahl von Kriterien an, die – bezogen auf riesige Datenbestände – die Ziele: Bewerten, Auswählen und Mitgestalten verfolgen. Im Zusammenhang mit Informationsräumen kommt den Kategorien Strukturieren, Navigieren und zielgerichtetes Interagieren eine große Bedeutung zu. Auf einem fachlich geprägten Hintergrund wird eine aktive Komplexitätsreduktion und damit -beherrschung mit informatischen Methoden und Konzepten eingeübt.

Die Zieldimension Befähigung und Reflexion zur Kollaboration ist bezüglich neuer Formen technisch gestützter Kommunikations- und Interaktionsprozesse (nicht nur) im Bereich der Arbeitsprozesse als weitere Schlüsselqualifikation auszuweisen. Die Notwendigkeit der produktiven, zielgerichteten Zusammenarbeit mit anderen Menschen in sozialer Verantwortung setzt die Befähigung zur Kollaboration unter den Aspekten Kommunikation, Kooperation, Koordination voraus. Die damit verbundenen Fähigkeiten ermöglichen ein reflektiertes und produktives Umgehen mit kollaborationsunterstützenden Informatiksystemen sowohl als Medium für die Interaktion wie auch als aktiver Assistent für die Arbeit.

Die Voraussetzung für „life long learning“ im Sinne des lebensbegleitenden Lernens sind in der Schule als Bestandteil des Informatikunterrichts zu schaffen.

Im fachlich orientierten Fundamentum der unterrichtlichen Umsetzung besteht in der Auseinandersetzung mit den Wirkprinzipien typischer, lerngruppenbezogener Vertreter von Informatiksystemen die Möglichkeit, behutsam am konkreten Modell in die Fachsystematik einzuführen. Das Ziel, die Kenntnis informatischer Aufgabenklassen anzulegen, muss auf didaktisch reduziertem Niveau vorbereitet werden. Dabei ist vor allem von einer vorschnellen wissenschaftlich exakten Bezeichnungsweise abzuraten, ohne dabei aber fachlich falsche Aussagen zu provozieren. Beispielsweise können die Aufgabenklassen mit *Lösbar*, *nicht lösbar*, *mit enormem Aufwand lösbar* bezeichnet werden.

Auf dem Hintergrund der Entwicklung der Informatik zu einer Schlüsselwissenschaft gilt es, das Bild der Wissenschaft im gesellschaftlichen Kontext zu vermitteln. Das Ziel besteht darin, dass die Gestaltungsmöglichkeiten für das Individuum erkannt werden und Gestaltungsfähigkeiten exemplarisch im Kontext der vorgestellten Elemente des grundlegenden Informatikunterrichts ausgebildet werden.

Im Zusammenhang mit dem Bild der Wissenschaft Informatik bei Schülerinnen nimmt der Autor wissenschaftliche Untersuchungen vor, deren Ergebnisse über [http://ddi.cs.uni-dortmund.de/gruppe/humbert\\_html](http://ddi.cs.uni-dortmund.de/gruppe/humbert_html) zugänglich gemacht werden. Erste Ergebnisse wurden mit [Humbert 2000c] vorgelegt: <http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Humbert.pdf>.

### 3.3.2 Oberstufe

Durch KMK-Beschlüsse [Kultusministerkonferenz 1999] sind die Rahmenbedingungen der gymnasialen Oberstufe wesentlich verändert worden. Die Neuorientierung *frisst* die Fächer, mit denen keine Pflichtbindung für das Abitur abgedeckt werden kann. Dies gilt für das Schulfach Informatik nach dem aktuellen Stand, wird aber möglicherweise im laufenden Jahr 2000 korrigiert - siehe Kasten in 3.2.

Ausgehend von der aktuellen Situation, d. h. ohne die oben angegebenen Voraussetzungen bei den Schülerinnen, besteht die Notwendigkeit, grundlegende Kenntnisse und Fähigkeiten bei den Schülerinnen zu entwickeln, damit darauf Bezug nehmend die Herausbildung informatischer Kompetenzen realisiert werden kann.

- **Variante 1: Kombikurse Informatik**<sup>20</sup>

Die Schulen haben die Möglichkeit, durch sogenannte Kombi[nations]kurse im mathematisch-naturwissenschaftlichen Bereich (z. B. Informatik und Physik, Informatik und Biologie, Informatik und Chemie, jeweils mit zwei Unterrichtsstunden und curricular aufeinander bezogen) die Informatik „ins Boot“ zu nehmen. Von Seiten der Informatik besteht - zumindestens bei den in geltenden Richtlinien in Nordrhein-Westfalen - die Offenheit, die für eine curriculare Abstimmung für fachübergreifende Ansätze notwendig sein wird.

In den Kombikursen ist mit vier Unterrichtsstunden pro Woche sowohl Informatik wie auch das jeweilige Koppelfach curricular zu erfüllen.

- **Variante 2: Schulprofile mit Informatik**

Es ist möglich, dass Schulen eine Profilbildung für die Sekundarstufe II realisieren: Dies bedeutet, dass schulisch inhaltliche, curriculare, aber auch methodische Schwerpunktsetzungen vorgenommen werden. Schulen bilden z. B. drei Oberstufenschwerpunkte, von denen ein Schwerpunkt mit einem ausdrücklich auf die Informatik ausgerichteten Profil gebildet wird.

Mit den Vorschlägen für einzelne Bausteine, die im zweiten Kapitel gemacht wurden, können Kurse in der Sekundarstufe II im 11. Jahrgang gestaltet werden.

Als Leitlinien eines solchen Spiralcurriculums erwiesen sich folgende Schwerpunkte als motivierend und in der Umsetzung erfolgreich:

- Der experimentelle Charakter der Repräsentation von theoretischen Elementen (vgl. [Humbert 1999a, S. 179ff]).
- Informatische Grundlagen der Nutzung informationstechnisch unterstützter Gruppenarbeit<sup>21</sup> mit informatischen Kernthemen.
- Die Ablösung der strukturierten Programmierung durch die objektorientierte Modellierung.
- Der Einsatz alternativer Problemlösungsmethoden der Informatik zur Förderung von Denkstrukturen, die nicht nur zur Bearbeitung von informatikspezifischen Fragestellungen genutzt werden.

---

<sup>20</sup>Für die Umsetzung in Kombikursen liegen bisher keine Erfahrungen vor. Die Schulprofilvariante wird in einigen gymnasialen Oberstufen erprobt.

<sup>21</sup>CSCW-Systeme - die Nutzung beschränkt sich aber nicht auf die Informatik

In dem folgenden Diagramm wird eine strukturierende Grobübersicht einer aktuellen Kursgestaltung des Autors vorgenommen. Da keine Vorkenntnisse bei den Schülern zu Beginn des 11. Jahrgangs vorausgesetzt werden können, besteht die Notwendigkeit, die unabdingbaren Vorbereitungen integriert in den Kursverlauf einzubinden. Die unterrichtliche Beschäftigung mit dem Thema CSCL (Computer Supported Cooperative Learning) kann nur integriert erarbeitet werden. Für die folgenden Jahrgänge besteht die Notwendigkeit der Integration konkreter Anwendungsbezüge für andere Unterrichtsfächer im Sinne fachüberschreitender Informatik-Qualifikationen. Beide Elemente werden damit in die fachsystematische Erarbeitung von Informatikinhalten eingebunden.

11. Jhg.	objektorientiertes Modellieren	Rechnernetze und verteilte Systeme elektronisches Publizieren	Computer Supported Cooperative Learning
12. Jhg.	Fachsystematik	Anwendungsbereich	Computer Supported Cooperative Learning
13. Jhg.	Fachsystematik	Anwendungsbereich	Computer Supported Cooperative Learning

## 4 Referenziertes Material

Auf folgende Quellen wird im html-Dokument Bezug genommen: [Peschke 1991], [Oestereich 1998], [Wasserschaff 1995].

### Literatur

[AG Neue Medien in der GEW NRW 1989] AG NEUE MEDIEN IN DER GEW NRW: Informationstechnische Grundbildung - aber wie? In: *FIFF-Kommunikation* 6. (1989), Nr. 1, S. 28–31. – Arbeitsgruppe Neue Medien im Referat Erziehungswissenschaften der GEW Nordrhein-Westfalen – Originalbeitrag im Heft 17/1988 Neue Deutsche Schule

[Appelrath u. a. 1998] APPELRATH, Hans-Jürgen ; BOLES, Dietrich ; CLAUS, Volker ; WEGENER, Ingo: *Starthilfe Informatik*. Stuttgart : B. G. Teubner, November 1998

- [Arlt und Koerber 1980] ARLT, Wolfgang ; KOERBER, Bernhard: Der Berliner Modellversuch zur Integration eines anwendungsorientierten Informatikunterrichts in der Sekundarstufe I. In: SCHAUER, Helmut (Hrsg.) ; TAUBER, Michael J. (Hrsg.): *Informatik in der Schule: Ergebnisse der Passauer Tagung*. München : Oldenbourg Verlag, 1980 (Schriftenreihe der Österreichischen Computer-Gesellschaft Bd. 7), S. 82–109
- [Balzert 1983] BALZERT, Helmut: *Informatik: 1. Vom Problem zum Programm – Hauptband*. 2. Aufl. München : Hueber-Holzmann Verlag, 1983. – 1. Aufl. 1976
- [Baumann 1991] BAUMANN, Rüdiger: *Prolog. Einführungskurs*. Stuttgart : Klett-Schulbuchverlag, 1991
- [Baumann 1996] BAUMANN, Rüdiger: *Didaktik der Informatik*. 2. völlig neu bearbeitete Aufl. Stuttgart : Klett-Schulbuchverlag, 1996
- [Baumert u. a. 1998] BAUMERT, Jürgen ; LEHMANN, Rainer ; LEHRKE, Manfred ; SCHMITZ, Bernd ; CLAUSEN, Marten ; HOSENFELD, Ingmar ; KÖLLER, Olaf ; NEUBRAND, Johanna: *TIMSS - Mathematisch-Naturwissenschaftlicher Unterricht im internationalen Vergleich - Deskriptive Befunde*. Berlin, Kiel : Max-Planck-Institut für Bildungsforschung, 1998
- [Berger 1997] BERGER, Peter: Das 'Computer-Weltbild' von Lehrern. In: **[Hoppe und Luther 1997]**, S. 27–39
- [Bosler 1992] BOSLER, Ulrich u. a. (Hrsg.): *Schulcomputerjahrbuch '93/94*. Hannover, Stuttgart : Metzler Schulbuch und B.G. Teubner, 1992
- [Brauer u. a. 1980] BRAUER, Wilfried ; HAACKE, Wolfhart ; MÜNCH, Siegfried ; GESELLSCHAFT FÜR MATHEMATIK UND DATENVERARBEITUNG MBH (Hrsg.): *Studien- und Forschungsführer Informatik*. 4. Ausgabe. Sankt Augustin/Bonn : GMD, 1980
- [Brauer u. a. 1989] BRAUER, Wilfried ; HAACKE, Wolfhart ; MÜNCH, Siegfried ; GESELLSCHAFT FÜR INFORMATIK E. V. (Hrsg.): *Studien- und Forschungsführer Informatik*. 2. neubearbeitete und erweiterte Aufl. Berlin, Heidelberg : Springer, 1989
- [Breutmann 2000] BREUTMANN, Bernd: *Informatik im Kontext des Lebenslangen Lernens*. [http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Inf\\_LLL.pdf](http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Inf_LLL.pdf). Juli 2000. – Tagungsbeitrag: Informatik - Ausbildung und Beruf, Fachhochschule Würzburg, 25.-27. Oktober 2000 Gesellschaft für Informatik e.V., Fachbereich 7 – Ausbildung und Beruf, Fachausschuss 7.1 – Informatik in Studiengängen an Hochschulen
- [Burkert und Peschke 1991] BURKERT, Jürgen ; PESCHKE, Rudolf ; HESSISCHES INSTITUT FÜR BILDUNGSPLANUNG UND SCHULENTWICKLUNG (HIBS) (Hrsg.): *Weiterentwicklung des Informatikunterrichts – Folgerungen aus der Sicht von Lehrerbildung und Wissenschaft*. Bd. 16. Wiesbaden : HIBS, 1991
- [Claus 1975] CLAUS, Volker: *Einführung in die Informatik*. Stuttgart : Teubner, 1975
- [Claus 1991] CLAUS, Volker: Die Rolle der Sprache – Anforderungen an den Informatikunterricht. In: **[Burkert und Peschke 1991]**, S. 148–158
- [Coy 1992] COY, Wolfgang: Einleitung: Informatik – Eine Disziplin im Umbruch? In: **[Coy u. a. 1992]**, S. 1–9
- [Coy u. a. 1992] COY, Wolfgang (Hrsg.) ; NAKE, Frieder (Hrsg.) ; PFLÜGER, Jörg-Martin (Hrsg.) ; ROLF, Arno (Hrsg.) ; SEETZEN, Jürgen (Hrsg.) ; SIEFKES, Dirk (Hrsg.) ; STRANSFELD, Reinhard (Hrsg.): *Sichtweisen der Informatik*. Braunschweig : Vieweg Verlag, 1992 (Theorie der Informatik)
- [von Cube 1960] CUBE, Felix von: *Allgemeinbildung oder produktive Einseitigkeit*. Stuttgart : Ernst Klett Verlag, 1960
- [CUU-Gruppe Gelsenkirchen 1973] CUU-GRUPPE GELSENKIRCHEN (Hrsg.): *Entwurf einer Unterrichtsstruktur des Faches Informatik an der Studienstufe der Gesamtschule Gelsenkirchen*. Gelsenkirchen : Gesamtschule, 1973
- [Czischke u. a. 1999] CZISCHKE, Jürgen ; DICK, Georg ; HILDEBRECHT, Horst ; HUMBERT, Ludger ; UEDING, Werner ; WALLOS, Klaus ; LANDESINSTITUT FÜR SCHULE UND WEITERBILDUNG (Hrsg.): *Von Stiften und Mäusen*. 1. Aufl. Bönen : DruckVerlag Kettler GmbH, 1999

- [Dijkstra 1968] DIJKSTRA, Edsger W.: Go To Statement Considered Harmful. In: *Communications of the ACM* 11 (1968), March, Nr. 3, S. 147–148
- [Dijkstra 1989] DIJKSTRA, Edsger W.: On the cruelty of Really Teaching Computing Science. In: *Communications of the ACM* 32 (1989), S. 1398–1404
- [Duit 1995] DUIT, Reinders: Zur Rolle der konstruktivistischen Sichtweise in der naturwissenschaftsdidaktischen Lehr- und Lernforschung. In: *Zeitschrift für Päd.* 41 (1995), Nr. 6, S. 905–923
- [Fachausschuß „Ausbildung“ der Gesellschaft für Informatik 1983] FACHAUSSCHUSS „AUSBILDUNG“ DER GESELLSCHAFT FÜR INFORMATIK. *Mitteilungen der Fachgruppe „Informatik in der Schule“*. März 1983
- [Fakultätentag Informatik 1999] FAKULTÄTENTAG INFORMATIK: *Beschluss der Fachkommission Informatik des AVI vom 10.9.1999*. <http://www.ft-informatik.de/protokolle/prot51-HH-AVIBeschluss.html>. November 1999. – Zimmermann, 23. Feb. 2000
- [Füller 1999] FÜLLER, Klaus: Objektorientiertes Programmieren in der Schulpraxis. In: **[Schwill 1999]**, S. 190–201
- [Gesellschaft für Informatik e. V. 1976] GESELLSCHAFT FÜR INFORMATIK E. V. (Hrsg.): *Zielsetzungen und Lerninhalte des Informatikunterrichts*. Stuttgart : Klett, 1976
- [Hans 1998] HANS, Ursula. *Prolog im Sommersemester 1998*. <http://pikas.inf.tu-dresden.de/~hans/lehre/prolog98/prolog98.html>. 1998
- [Hollenstein 1998] HOLLENSTEIN, Armin: Kognitive Aspekte Sozialen Lernens. In: *Pädagogisches Handeln – Wissenschaft und Praxis im Dialog* (1998), Dezember, S. 6. – Vorabdruck – 10. Dezember 1998
- [Hoppe und Luther 1997] HOPPE, Heinz U. (Hrsg.) ; LUTHER, Wolfram (Hrsg.): *Informatik und Lernen in der Informationsgesellschaft*. Berlin, Heidelberg : Springer, September 1997 (Informatik aktuell)
- [Hubwieser 1998] HUBWIESER, Peter: *Didaktik der Informatik I - Vorlesungsskript*. München : Technische Universität München, 1998. – Fakultät für Informatik <http://www-schulen.informatik.tu-muenchen.de/hubwiese/publist.html>
- [Hubwieser und Broy 1997] HUBWIESER, Peter ; BROY, Manfred: Grundlegende Konzepte von Informations- und Kommunikationssystemen für den Informatikunterricht. In: **[Hoppe und Luther 1997]**, S. 40–50
- [Humbert 1998a] HUMBERT, Ludger. *CSCW im Unterricht – Unterrichtsentwurf für eine einführende Unterrichtsstunde*. <http://in.Hagen.de/humbert/vortraege/Unterrichtsentwurf.CSCW.ps.gz>. Juni 1998a
- [Humbert 1998b] HUMBERT, Ludger: Das Internet - Möglichkeiten zur Nutzung in der zweiten Ausbildungsphase. In: *Seminar - Lehrerbildung und Schule* (1998), November, Nr. 2, S. 41–50. – <http://in.Hagen.de/humbert/vortraege/seminar/welcome.html>
- [Humbert 1998c] HUMBERT, Ludger. *Kleine, subjektive Geschichte der Verbindung der Gesamtschule Haspe (Hagen) mit dem Internet*. <http://in.hagen.de/humbert/vortraege/GE-Haspe/Internet.Geschichte.html>. Dezember 1998c
- [Humbert 1999a] HUMBERT, Ludger: Grundkonzepte der Informatik und ihre Umsetzung im Informatikunterricht. In: **[Schwill 1999]**, S. 175–189
- [Humbert 1999b] HUMBERT, Ludger: Kollaboratives Lernen - Gruppenarbeit im Informatikunterricht. In: *LOG IN* 19 (1999), Nr. 3/4, S. 54–59
- [Humbert 2000a] HUMBERT, Ludger. *Bericht zur Lehrerausbildung Informatik*. <http://koenigstein.inf.tu-dresden.de/00/humbert2.html>. März 2000a
- [Humbert 2000b] HUMBERT, Ludger: Ein System zur Unterstützung kollaborativen Lernens – Computerunterstützte Gruppenarbeit in der Sekundarstufe II. In: *Computer und Unterricht* (2000), August, Nr. 39, S. 28–31



- [Humbert 2000c] HUMBERT, Ludger: *Theoretischer und empirischer Vergleich zum Bild der Wissenschaft Informatik in der Schule*. <http://netlab01.fh-wuerzburg.de/Tagung/IAB2000/deutsch/info/Plenum1/Humbert.pdf>. Juli 2000c. – Tagungsbeitrag: Informatik - Ausbildung und Beruf, Fachhochschule Würzburg, 25.-27. Oktober 2000 Gesellschaft für Informatik e. V., Fachbereich 7 – Ausbildung und Beruf, Fachausschuss 7.1 – Informatik in Studiengängen an Hochschulen
- [König 1993] KÖNIG, Gerhard: Informatikunterricht aus der Sicht der Hochschule - Ergebnisse einer Umfrage. In: *Zentralblatt für Didaktik der Mathematik* 25 (1993), Nr. 1, S. 1–8
- [Krabbel und Kuhlmann 1994] KRABEL, Anita ; KUHLMANN, Bettina ; BRUNNSTEIN, Klaus (Hrsg.) ; OBERQUELLE, Horst (Hrsg.): *Zur Selbstverständnis-Diskussion in der Informatik / Universität Hamburg, Fachbereich Informatik*. D-22527 Hamburg, Vogt-Kölln-Strasse 30 : Uni Hamburg, FB IF, Mai 1994 ( Bericht FBI-HH-B-169/94). – Forschungsbericht
- [Kroha 1997] KROHA, Petr: *Softwaretechnologie*. 1. Aufl. München : Prentice-Hall, 1997
- [Kultusministerkonferenz 1999] KULTUSMINISTERKONFERENZ: *Vereinbarung zur Gestaltung der gymnasialen Oberstufe in der Sekundarstufe II*. <http://www.kmk.org>. Oktober 1999. – Beschluss der KMK vom 7. Juli 1972 i. d. F. vom 22. Oktober 1999
- [Lehmann 1993] LEHMANN, Gabriele: Methoden der Zielbestimmung für die Informatische Bildung. In: TROITZSCH, Klaus G. (Hrsg.): *Informatik als Schlüssel zur Qualifikation*. Berlin, Heidelberg : Springer, 1993 (Informatik aktuell), S. 348–354
- [Loethe 1997] LOETHE, Herbert M.: Computer integration in the mathematics study of pre-service teacher education – experiences with Project CIMS. In: PASSEY, Don (Hrsg.) ; SAMWAYS, Brain (Hrsg.): *Information Technology – Supporting change through teacher education*. London : Chapman & Hall, 1997, S. 238–244
- [Löwis und Fischbeck 1997] LÖWIS, Martin v. ; FISCHBECK, Nils: *Das Python-Buch: Referenz der objektorientierten Skriptsprache für GUIs und Netzwerke*. 1. Aufl. Bonn : Addison-Wesley, 1997
- [Lutz und Ascher 2000] LUTZ, Mark ; ASCHER, David: *Einführung in Python*. 1. Aufl. Köln : O'Reilly, März 2000. – Übersetzung von „Learning Python“ 1999, O'Reilly Deutsche Übersetzung von Christian Tismer und Karin Balster
- [Meißner 1975] MEISSNER: Informatik I, II. In: *Der Mathematikunterricht* 18, 21 (1971 und 1975), Nr. H. 1, H. 5
- [MSWWF 1999] MSWWF (Hrsg.): *Richtlinien und Lehrpläne für die Sekundarstufe II - Gymnasium/Gesamtschule in Nordrhein-Westfalen - Informatik*. 1. Aufl. Frechen : Ritterbach Verlag, Juni 1999. – MSWWF (Ministerium für Schule und Weiterbildung, Wissenschaft und Forschung des Landes Nordrhein-Westfalen)
- [MSWWF 2000] MSWWF (Hrsg.): *Schriftenreihe Schule in NRW*. Bd. 9032: *Zukunft des Lehrens – Lernen für die Zukunft: Neue Medien in der Lehrerbildung – Rahmenkonzept*. Frechen : Ritterbach Verlag, März 2000. – Materialien Lehrerbildung Heft 9032 — MSWWF (Ministerium für Schule und Weiterbildung, Wissenschaft und Forschung des Landes Nordrhein-Westfalen)
- [Müller 1992] MÜLLER, Klaus: Objektorientierte Programmierung. In: **[Bosler 1992]**, S. 180–189
- [Nora und Minc 1979] NORA, Simon ; MINC, Alain: *Die Informatisierung der Gesellschaft*. Frankfurt a. M. : Campus Verlag, 1979 (Veröffentlichungen der Gesellschaft für Mathematik und Datenverarbeitung)
- [Oestereich 1998] OESTEREICH, Bernd: *Objektorientierte Softwareentwicklung - Analyse und Design mit der Unified Modeling Language*. 4. aktualisierte Aufl. München : Oldenbourg Verlag, 1998
- [Padawitz 1995] PADAWITZ, Peter: *Programmierung II - Sommersemester*. Dortmund : Universität Dortmund, 1995. – Vorlesungsskript - FB Informatik, LS 5, Universität Dortmund
- [Padawitz 1998] PADAWITZ, Peter: *Einführung ins funktionale Programmieren – Vorlesungsskriptum*. Dortmund : Universität Dortmund, 1998. – LV Grundlagen und Methoden funktionaler Programmierung <http://ls5-www.cs.uni-dortmund.de/padawitz.html>

- [Parnas 1990] PARNAS, David L.: Education for Computing Professionals. In: *IEEE Computer* 23 (1990), Nr. 1, S. 17–22
- [Peschke 1991] PESCHKE, Rudolf: Stand der informationstechnischen Bildung und Folgen für den Informatikunterricht. In: **[Burkert und Peschke 1991]**, S. 121–138
- [Rolf 1992] ROLF, Arno: Grundlagen einer Theorie der Informatik: Sichtwechsel - Informatik als Gestaltungswissenschaft. In: **[Coy u. a. 1992]**, S. 33–47
- [Schubert 1992] SCHUBERT, Sigrid: Logische Programmierung. In: **[Bosler 1992]**, S. 171–179
- [Schubert 2000] SCHUBERT, Sigrid: *Skriptum zur Vorlesung Didaktik der Informatik (Sekundarstufe II)*. <http://ddi.cs.uni-dortmund.de/lehre/grundstudium/skript.html>. Januar 2000. – Universität Dortmund, Didaktik der Informatik
- [Schwill 1993] SCHWILL, Andreas: Fundamentale Ideen der Informatik. In: *Zentralblatt für Didaktik der Mathematik* 25 (1993), Nr. 1, S. 20–31. – bearbeitete Fassung in Engl. in: *EATCS-Bulletin* No. 53 (1994), S. 274–295, <http://www.didaktik.cs.uni-potsdam.de/Forschung/Schriften/EATCS.pdf>
- [Schwill 1999] SCHWILL, Andreas (Hrsg.): *Informatik und Schule - Fachspezifische und fachübergreifende didaktische Konzepte*. Berlin : Springer, September 1999 (Informatik aktuell)
- [Staudte 1992] STAUDTE, Rainer: Funktionale Programmierung. In: **[Bosler 1992]**, S. 160–171
- [Wasserschaff 1995] WASSERSCHAFF, Markus: *Maßgeschneiderte Benutzerschnittstellen zur Unterstützung kooperativer Arbeit*. August 1995. – Diplomarbeit, Universität Bonn - Institut für Informatik <http://mats.gmd.de/markusw/Papers/Papers.html>