

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/266454547>

# Fachliche Orientierung des Informatikunterrichts in der Sekundarstufe II

## Article

CITATIONS

2

READS

17

## 2 authors:



**Ludger Humbert**

Bergische Universität Wuppertal

67 PUBLICATIONS 69 CITATIONS

[SEE PROFILE](#)



**Sigrid Schubert**

Universität Siegen

100 PUBLICATIONS 349 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Veranstaltung »Didaktik der Informatik« -- Sommersemester 2015 [View project](#)



#PflichtfachInformatik – Informatik an Grundschulen – #cs4all – informatics at primary level [View project](#)

**Fachliche Orientierung des  
Informatikunterrichts  
in der Sekundarstufe II**

Ludger Humbert, Sigrid Schubert  
Didaktik der Informatik  
Universität Dortmund

Report Nr. 771  
Februar 2002



# **Fachliche Orientierung des Informatikunterrichts in der Sekundarstufe II**

**Ludger Humbert, Sigrid Schubert**

Didaktik der Informatik

Universität Dortmund

Report Nr. 771

Februar 2002

## Zusammenfassung

Die Fruchtbarmachung der grundlegenden Konzepte der Fachwissenschaft Informatik für die Umsetzung in allgemein bildenden Prozessen ist eine der originären Aufgaben der Didaktik der Informatik. Seit Beginn der Arbeit der Fachgruppe „Didaktik der Informatik“ an der Universität Dortmund (1998) werden von den Autoren Untersuchungen durchgeführt, um die Ausgestaltung von Basiskonzepten der Informatik zu einer schlüssigen, offenen Lehrkonzeption zu entwickeln und zu evaluieren. Dazu müssen gestaltende Tätigkeiten auf der Grundlage fachwissenschaftlicher Erkenntnisse für Lehr- und Lernkonzepte durchgeführt werden.

In der inzwischen über 30jährigen Geschichte des Schulfachs Informatik in bundesdeutschen Schulen wird im vorliegenden Forschungsbericht eine Analyse der zukunftsweisenden Ansätze aus dieser zurückliegenden Phase vorgenommen. Die Ergebnisse dieser Analyse münden in konkreten forschungsleitenden Fragestellungen. Die Untersuchung dieser Fragestellungen haben das Ziel, zu einer tragfähigen Konzeption modernen Informatikunterrichts in der allgemein bildenden Schule zu gelangen. Die in diesem Kontext gewonnenen und entwickelten Ansätze sind qualifiziert in einen Gesamtrahmen einzubinden, um dem allgemein bildenden Anspruch gerecht werden zu können. Dazu wird von den Autoren die Qualität der erzielten kognitiven Strategien und Verhaltensänderungen bei Schülerinnen untersucht. Die in diesen Untersuchungen zutage geförderten Ergebnisse werden weiteren Untersuchungen zugeführt, die darin bestehen, unterschiedliche Ansätze auf der Folie der gewonnenen Ergebnisse zu vergleichen. Aus diesem Untersuchungszyklus lassen sich Gestaltungsanforderungen entwickeln, die für den Aufbau Didaktischer Systeme, für die Gestaltung von Unterrichtssequenzen, aber insbesondere für die spiralförmige Schichtung curricularer Elemente in einem theoriegeleiteten Rahmen unverzichtbar sind.

Bei der Analyse tritt zu Tage, dass häufig ein Sammelsurium von Inhalten als Struktur der Schulinformatik additiv zu curricularen Ketten verbunden wurde. Diese Struktur sollte überwunden werden, um eine fachlich solide Basis für eine Gesamtkonstruktion und zukunftsorientierte Konzeption in das Zentrum curricularer Anstrengungen zu rücken. Ein Untersuchungsgegenstand dieses Umfangs läßt sich nicht mit einer Forschungsarbeit adäquat bewältigen. Aus diesem Grund werden an vielen Stellen Fragen aufgeworfen, die im Rahmen weiterer Forschungen einer Beantwortung zugeführt werden sollten.

Mit dem Modulkonzept wird ein Ansatz vorgestellt, der die qualifizierte Schichtung inhaltlicher Elemente für das Schulfach Informatik in Abhängigkeit von den kognitiven Fähigkeiten der Schülerinnen in einem Spiralcurriculum ermöglicht. Damit wird der Beliebigkeit der Fachinhalte, die sich heute in der verschiedenen Schwerpunktsetzungen und in der unqualifizierten Kopplung von Informatik mit anderen Fächern – falsch als Integration bezeichnet – der Boden entzogen. Nur ein modulares Geflecht essenzieller Elemente, das den fachlichen, kognitiven und pädagogischen Anforderungen an eine moderne Bildung genügt, vermag auf Dauer die Situation des Schulfachs Informatik nachhaltig qualifiziert zu beeinflussen.

## Inhaltsverzeichnis

<b>Zusammenfassung</b> . . . . .	<b>ii</b>
<b>Inhaltsverzeichnis</b> . . . . .	<b>iii</b>
<b>1 Motivation aufgrund der Anforderungen an die Schulinformatik</b> . . . . .	<b>1</b>
<b>2 Entwicklung von Grundkonzepten in der Wissenschaft Informatik</b> . . . . .	<b>4</b>
2.1 Bildungsschlussfolgerungen aus wissenschaftstheoretischen Positionen . . . . .	4
2.2 Fachdidaktische Empfehlungen aus Sicht der Fachwissenschaft . . . . .	6
<b>3 Entwicklung der Didaktik der Informatik für Schulen</b> . . . . .	<b>11</b>
3.1 Entwicklung in der Bundesrepublik Deutschland . . . . .	11
3.2 Internationale Diskussion . . . . .	20
<b>4 Das Modulkonzept</b> . . . . .	<b>23</b>
4.1 Wissenschaftliche Fragestellungen . . . . .	23
4.2 Zur konzeptionellen Gestaltung – Vorstellung der Module . . . . .	24
4.2.1 Informatiksysteme verstehen und verantwortlich nutzen . . . . .	25
4.2.2 Modellierung - zentrales Tätigkeitsfeld informatischer Arbeit . . . . .	25
4.3 Konkretisierung – Dokumentation von Kursbeispielen . . . . .	28
4.4 Untersetzung der Fragestellungen in Untersuchungsvorhaben . . . . .	29
<b>5 Ergebnisse – Perspektiven</b> . . . . .	<b>31</b>
<b>Anhang</b> . . . . .	<b>33</b>
<b>A Gestaltung eines Grundkurses Informatik (11., 12. Jhg.)</b> . . . . .	<b>33</b>
<b>B Ausgewählte Unterrichtsplanung und -durchführung</b> . . . . .	<b>36</b>
B.1 Objektorientierte Modellierung am Beispiel des Schülercomputerarbeitsplatzes . . . . .	36
B.2 Modellierung grundlegender Elemente für ein „Blinklicht“ . . . . .	37
B.3 Einführung in die Arbeit mit einem webbasierten CSCW zur Kursplanung . . . . .	38
B.4 Das Fachkonzept Keller - Realisierung als Klasse . . . . .	42
B.5 Nebenläufigkeit zur Realisierung der „Parallelität“ von Server und Klient . . . . .	45
<b>Literaturverzeichnis</b> . . . . .	<b>51</b>



## 1 Motivation aufgrund der Anforderungen an die Schulinformatik

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

Es ist bisher nicht gelungen, für das Fach Informatik in der Sekundarstufe I (Jahrgangsstufen 5 bis 10) allgemein bildender Schulen einen verbindlichen Lernort zu schaffen. Auf Grund der zunehmenden Durchdringung aller Lebensbereiche mit Informatiksystemen zeichnen sich zwei Entwicklungslinien ab. Zum einen sind grundlegende Informatikkenntnisse in den traditionellen Berufsfeldern unabdingbar und zum anderen ergeben sich verstärkt neue Berufsmöglichkeiten im Informatikbereich (vgl. exemplarisch [Dostal 2000]) für die spezifische Ausbildungsformen zu entwickeln sind. Die neuen Ausbildungsberufe im Informationstechnologie (IT)-Bereich sind ein Beispiel dafür. Für diese Berufe wurden im dualen Ausbildungssystem Ausbildungsgänge eingerichtet. Die formale Eingangsvoraussetzung besteht im erfolgreichen Abschluss der Sekundarstufe I. Bis zu diesem Zeitpunkt hat eine Schülerin keine schulisch verpflichtende informatische Grundqualifikation erworben. Heute ist Informatik u. a. gekennzeichnet durch die objektorientierte Konstruktion von Informatiksystemen. Der Begriff Informatiksysteme wird von den Autoren zur Bezeichnung der Einheit von Hard- und Software unter Betonung ihrer interaktiven Eigenschaften eingesetzt. Informatiksysteme werden unter stärkerer Berücksichtigung verteilter und vernetzter Systeme gestaltet. Diese beiden Aspekte stellen eine Umorientierung des informatischen Wissenschaftsverständnisses dar, die als Paradigmenwechsel bezeichnet werden kann. Im Laufe der Zeit wurden von der Didaktik der Informatik verschiedene, sich widersprechende Vorschläge zur Umsetzung informatischer Fachkonzepte empfohlen, die unzureichend Eingang in die Unterrichtspraxis des Schulfaches Informatik fanden.

Die Autoren vertreten den Standpunkt, dass der Paradigmenwechsel der Fachwissenschaft eine Neuorientierung des Informatikunterrichts bewirken muss. Bevor die Frage nach der Auswahl und Umsetzung dieser neuen Fachkonzepte beantwortet werden kann, muss die allgemein bildende Zielsetzung der Informatischen Bildung<sup>1</sup> in der sogenannten Informations- bzw. Wissensgesellschaft analysiert werden. Hier reichen die traditionellen Studien zur Allgemeinbildung nicht mehr aus. „Die Informationsgesellschaft soll neue Märkte und Arbeitsplätze bringen, das Leben bequemer machen, unser Wissen über die Welt vergrößern und vertiefen und die Teilhabe an der politischen Meinungsbildung sowie der Gestaltung unserer Lebensverhältnisse verbessern. Dazu müssen möglichst viele Menschen mit den verschiedenen technischen Systemen umgehen können. [...] Sonst werden die heute Ausgebildeten weder die späteren beruflichen Anforderungen erfüllen, noch ihre persönlichen Entwicklungschancen realisieren können. Unternehmen, die sich heute moderne, schlanke, flexible Organisationskonzepte geben, haben jetzt schon Umsetzungsprobleme, weil diese Fähigkeiten und andere soziale Kompetenzen zu wenig entwickelt sind.“ [Kubicek 1996] „Aufgabe der allgemein bildenden Schule muss es sein, allen Schülerinnen und Schülern unabhängig von ihrem Geschlecht, ihrer Herkunft und ihren sozialen Verhältnissen einen gleichberechtigten Zugang zu informatischen Denk- und Arbeitsweisen und modernen Informations- und Kommunikationstechniken zu öffnen, informatische Bildung zu vermitteln und damit auch auf lebenslanges Lernen, d. h. auf die Möglichkeiten der ständigen Wissensreorganisation, vorzubereiten. Informatische Bildung ist das Ergebnis von Lernprozessen, in denen Grundlagen, Methoden, Anwendungen, Arbeitsweisen und die gesellschaftliche Bedeutung von Informatiksystemen erschlossen werden. Dazu trägt insbesondere der Informatikunterricht in den Sekundarstufen I und II bei. Unterrichtsangebote, in denen interaktive Informatiksysteme als Werkzeug und Medium in anderen Fächern eingesetzt werden, gehören nur dann

<sup>1</sup>Die Autoren verwenden Informatische Bildung als Bezeichnung.



zur informatischen Bildung, wenn informatische Aspekte bewusst thematisiert werden. In allen Phasen der informatischen Bildung stellt die Informatik die Bezugswissenschaft dar“ [Gesellschaft für Informatik 2000, S. 376]. Die zur Zeit vorhandenen Phasen der Informatischen Allgemeinbildung werden im Folgenden analysiert, um ihren Beitrag für die einführend beschriebenen Bildungsanforderungen einzuschätzen.

### **Informationstechnische Grundbildung**

1984 beschloss die Bund-Länder-Kommission (BLK) die bundesweit verpflichtende Informationstechnische Grundbildung (ITG) in den Jahrgangsstufen 7-9 der Sekundarstufe I [BLK 1984]. Die damit verbundene Hoffnung – der qualifizierten Anwendung von Informatiksystemen durch die Schülerinnen in jedem Fachunterricht – konnten auf Grund der fehlenden Inhalte der Informatik nicht erfüllt werden. Genau dieser Fehler wird zur Zeit im Zusammenhang mit der breiten Anwendung des Internet in schulischen Lehr-Lern-Prozessen wiederholt, die eine fundierte Auseinandersetzung mit den Möglichkeiten und Grenzen der komplexen Informatiksysteme verlangt.

Wie bereits bei der ITG:

- sollen Schülerinnen mit Anwendungen selbstbestimmt arbeiten, ohne die fachlichen Grundlagen dafür zu besitzen.
- werden Lehrerinnen mit Bildungsaufgaben konfrontiert, bevor dafür empirisch erprobte Konzepte zur Verfügung stehen.

### **Informatikunterricht in der Sekundarstufe I**

In allen Bundesländern existieren im Wahlpflichtbereich des 9. und 10. Jahrgangs Informatikangebote. In diesen Kursen werden häufig Informatikanwendungen genutzt. Das informatische Modellieren und das Verständnis der Wirkprinzipien kommt zu kurz. Das gilt auch für das Fach Angewandte Informatik an Sächsischen Mittelschulen (Haupt- und Realschulen) von Jahrgangsstufe 7 bis 10.

Es gibt neue Entwicklungen:

- Im gymnasialen Plan Nordrhein-Westfalens besteht die Möglichkeit des objektorientierten Modellierens (vgl. [Kultusministerium des Landes Nordrhein-Westfalen 1993, S. 48]).
- Der Informatiklehrplan für alle Schularten in Mecklenburg-Vorpommern setzt das Spiralcurriculum mit vier Leitlinien um:
  - Umgang mit Information
  - Wirkprinzipien von Informatik-Systemen
  - Problemlösen mit Informatik-Systemen
  - Wechselwirkungen zwischen Informatik-Systemen und Individuum bzw. Gesellschaft

vgl. [Kultusministerium Mecklenburg-Vorpommern 1998, S. 19]

- Im gymnasialen Plan Bayerns wird künftig (2005/2006) informatisches Modellieren gefordert mit den Schwerpunkten Datenmodellierung, funktionale Modellierung, objektorientierte Modellierung.

Damit leistet ausschließlich Mecklenburg-Vorpommern einen verbindlichen Beitrag zur informatischen Bildung für alle Schülerinnen in der Sekundarstufe I, auf den die Sekundarstufe II aufbauen kann. In einigen Bundesländern (vgl. Bayern) fehlt das Angebot für bestimmte Schularten. In anderen Bundesländern

(vgl. Nordrhein-Westfalen) sind die Lerngegenstände so unterschiedlich, dass kein verbindendes Fundament erreicht wird. 2000 wurde ein „Gesamtkonzept der Informatischen Bildung an allgemein bildenden Schulen“ veröffentlicht [Gesellschaft für Informatik 2000]. Es liefert einen Orientierungsrahmen für Kompetenzen und Lerngegenstände, der einen großen Interpretationsspielraum läßt und mit konkreten Unterrichtskonzepten ausgestaltet werden muss. Ein Pflichtfach Informatik muss in der fünften Jahrgangsstufe einsetzen, damit andere Schulfächer auf grundlegende Kompetenzen aus einem obligatorischen Informatikunterricht zurückgreifen können.

### **Informatikkurse in der Sekundarstufe II (gymnasiale Oberstufe)**

Das Zusammentreffen von gesellschaftlichen Erfordernissen mit der Neugestaltung der gymnasialen Oberstufe führte ab 1972 zur Einführung von Grundkursen in der allgemein bildenden Sekundarstufe II. Nicht alle Bundesländer ermöglichen Leistungskurse. Die Schülerinnen erfüllen durch die Belegung eines Informatikkurses in vielen Bundesländern keine fachliche Pflichtbindung im naturwissenschaftlich-mathematisch-technischen Aufgabenfeld. Dies führt zu kleinen Kursen, die die durchschnittliche Kursgröße einer Schule belastet und deshalb nur unter schwierigen Bedingungen über mehrere Jahrgangsstufen erhalten werden können. 1976 veröffentlichte die Gesellschaft für Informatik (GI) erstmals Empfehlungen für die Ziele und Inhalte der Informatischen Bildung [Brauer u. a. 1976]. Es gab einen breiten Konsens für den Schwerpunkt „Algorithmen und Datenstrukturen“, der darauf beruhte, dass die Informatik als Wissenschaft noch wenige Grundkonzepte vorzuweisen hatte. Die Aufgabenklassen, die Anfängerinnen in Hochschule und Schule analysierten und gestalteten, lagen sehr dicht beisammen und wiesen noch eine erstaunliche Nähe zur professionellen Tätigkeit in diesem Berufsfeld auf. 1989 wurde von „Der Krise des Informatikunterrichts“ [Peschke 1989] gesprochen, da erkannt war, dass die Informatische Bildung nicht über eine Abbilddidaktik aus der Fachwissenschaft abgeleitet werden kann. Seitdem konnte kein neuer Konsens über die im Informatikunterricht anzueignenden Grundkonzepte hergestellt werden.

## 2 Entwicklung von Grundkonzepten in der Wissenschaft Informatik

### 2.1 Bildungsschlussfolgerungen aus wissenschaftstheoretischen Positionen

Notwendig erscheint angesichts der sich konsolidierenden Fachwissenschaft die Herausarbeitung der langfristig gültigen Konzepte, die Verbindung mit Fachmethoden, die kognitiven Bedürfnissen gerecht werden. Unter dem Dach der Informatik finden sich heute noch verschiedene, widersprüchliche wissenschaftstheoretische Sichtweisen. Begründet ist das einerseits in der erfolgreichen Ablösung von den Disziplinen Mathematik und Elektrotechnik, die die Informatik hervorbrachten, und andererseits in der sich klarer abzeichnenden Brücke zu den Geisteswissenschaften. Erhalten blieb der Charakter einer Strukturwissenschaft analog zur Mathematik, die aber andere Denkweisen und Methoden hervorbrachte. Neu ist das Verständnis für soziotechnische Systeme und die Anwendung partizipativer Konzepte in der Softwareentwicklung. Seit 1970 wurden unterschiedliche Auffassungen von der Informatik als Wissenschaft veröffentlicht. Die folgenden zentralen Schlüsselbegriffe ermöglichen eine grobe Einordnung verschiedener Auffassungen.

WEIZSÄCKER spricht von zwei Strukturwissenschaften, der Mathematik und der Informatik [von Weizsäcker 1971]. Damit verbindet er den Anspruch an die Informatik, für andere Disziplinen die automatisierte Informationsverarbeitung bereitzustellen. Geistige Techniken der Strukturierung, wie z. B. Hierarchisieren und Modularisieren, sind unabhängig von der technischen Realisierung wertvolle Methoden zur Beherrschung von Komplexität.

„Im Vordergrund stehen prinzipielle Verfahren und nicht spezielle Realisierungen [...]. Die Inhalte der Informatik sind daher vorwiegend logischer Natur und maschinenunabhängig.“ [Claus 1975, S. 11]. CLAUS bestärkt damit die Sichtweise WEIZSÄCKERS auf die Informatik und betont den Wissenschaftsschwerpunkt von „Algorithmen und Datenstrukturen“, die mit formalen Sprachen beschrieben und mit logischen Kalkülen ausgeführt werden.

GENRICH und PETRI kommen ausgehend vom Informationsbegriff der Kybernetik zur Sicht auf die Informatik als „Wissenschaft vom streng geregelten Informationsfluß.“ [Genrich 1975, Petri 1983]. Damit wird die Beschreibung von Prozessen einschliesslich der gesellschaftlichen Nützlichkeitsbetrachtung in den Mittelpunkt gerückt. Zur Bewertung von Informatiksystemen tragen dann nicht ausschließlich Strukturierungsqualität und algorithmische Effizienz, sondern auch die Einbettung in die sozialen Anwendungsprozesse bei.

NYGAARD : „When it is argued that informatics is a formal discipline only, then "according to such a definition, the impact of an information system upon the social structure of which it is a part, is outside (the field of study of) informatics. Also case studies of how data processing actually is carried out in specific organizations fall outside informatics in this narrow sense." [Håndlykken und Nygaard 1981]<sup>2</sup>

Damit verdeutlicht NYGAARD, dass eine zu enge Sicht auf die Informatik aufgehoben werden muss.

---

<sup>2</sup>zitiert nach [Nygaard 1986, S. 189]

Mit ihrer wachsenden gesellschaftlichen Bedeutung wurde die Informatik zum Gegenstand wissenschaftstheoretischer Vergleiche, die weit über die innerdisziplinären Aspekte hinausreichen. So charakterisierte CAPURRO die „Informatik als hermeneutische Disziplin“ [Capurro 1990, S. 315] mit der Aufgabe der „*technischen* Gestaltung [...] menschlicher] Interaktionen in der Welt“ [Capurro 1990, S. 317].

Diese Einschätzung geht vielen Informatikerinnen zu weit, da sie sich von den damit verbundenen Konsequenzen überfordert fühlen. Gerade für die Didaktik der Informatik eröffnet diese philosophische Dimension Chancen für Synergieeffekte zwischen Unterrichtsfächern.

Mit FLOYD gelang die produktive Verbindung der ursprünglich gegensätzlichen Positionen. „That means, it views itself as a formal and an engineering science, relying strongly on the traditional scientific paradigm [...]“. [Floyd 1992, S. 19]

Eine ausführlichere Darstellung wird einigen Positionen gewidmet, die im Jahr 1992 veröffentlicht wurden, da sie eine wichtige Phase in der Standortbestimmung der Informatik einleiten.

COY: „Aufgabe der Informatik ist also die Analyse von Arbeitsprozessen und ihre konstruktive, maschinelle Unterstützung. Nicht die Maschine, sondern die Organisation und Gestaltung von Arbeitsplätzen steht als wesentliche Aufgabe im Mittelpunkt der Informatik. Die Gestaltung der Maschinen, der Hardware und der Software ist dieser primären Aufgabe untergeordnet. Informatik ist also nicht ‚Computerwissenschaft‘. An dieser Stelle zeigt sich deutlich, daß sich Informatik von Nachrichten- oder Informationstechnik in ihrer Ausrichtung wesentlich unterscheidet; der Kern der Unterscheidung liegt in der viel engeren Kopplung der Informatik an reale Arbeitsprozesse.“ [Coy 1992, S. 18f]

Die Autoren stimmen mit dieser Position nur insofern überein, als Informatik nicht nur als Strukturwissenschaft und technische Wissenschaft betrachtet werden kann, sondern mit diesen Prinzipien und Methoden zusammen, der soziotechnische Kontext in der Informatikausbildung zu thematisieren ist. Im Anhang werden dazu Unterrichtsbeispiele dokumentiert. Entschieden abgelehnt wird von den Autoren die Argumentation von COY, die die Informatik den Sozialwissenschaften zuordnet. Im Rahmen der Allgemeinbildung wurde damit bisher das Pflichtfach Informatik für alle Schülerinnen abgelehnt und die fachlich fundierte Ausbildung von Lehrerinnen aufgehalten.

LUFT charakterisiert Informatik als eine Disziplin, die „[...] im Hinblick auf Entwurf und Gestaltung der Architektur weitaus näher steht als der Elektro- und Nachrichtentechnik.“ [Luft 1992, S. 50]

Diese Position ist durch die Erfolge von Entwurfsmustern in der Softwaretechnik ausgesprochen aktuell. Zu prüfen bleibt die kognitive Wirksamkeit von Grundmustern, die im Rahmen der Informatikausbildung Schülerinnen beim Wissenstransfer unterstützen sollen.

ACM veröffentlichte 1997 und 1999 Artikel zu den potenziellen Möglichkeiten der Informatik [Denning und Metcalfe 1997, Denning 1999b]. Aus dieser Vielfalt sei DENNING zitiert

„[...] Computers have given us new ways of thinking about machines, communications, organizations, societies, countries, and economies. [...] A growing number of educators, for example, say that there is much more to learning than transferring information; they say the phenomenon of embodied knowledge, learned

through practice and involvement with other people, is a process that cannot be understood simply as information transfer.“ [Denning 1999a, S. 6f der Online-Fassung]

Bedauerlich ist die Bindung an das Gerät Computer. Nach Meinung der Autoren charakterisiert die Aussage eher die Denkweisen der Informatik als deren Abbildung auf die informationsverarbeitenden Maschinen. Das Zitat ist hilfreich, um Bildungskonzepte unter der Überschrift Medienkompetenz auf ihre Wirksamkeit bezüglich der informatischen Basiskonzepte zu überprüfen.

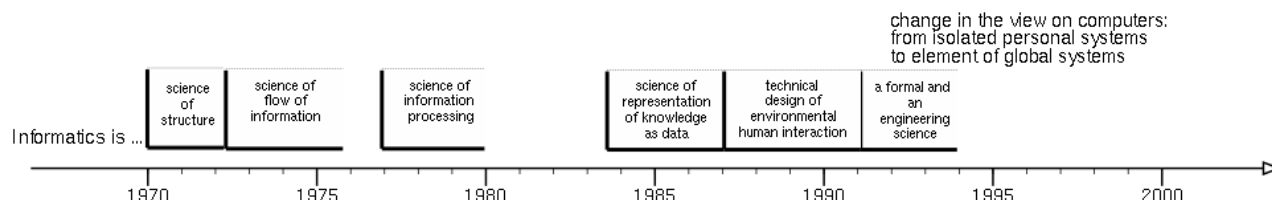


Abbildung 1: Zeitleiste zu [Innen-] Sichten der Informatik

## 2.2 Fachdidaktische Empfehlungen aus Sicht der Fachwissenschaft

Da noch keine Forschungsgruppen zur Didaktik der Informatik existierten, stammen die ersten wissenschaftlichen Studien zur Lehrdisziplin der Informatik von Fachwissenschaftlern. Die Autoren möchten hier nicht den gesamten historischen Weg nachzeichnen, sondern wählten solche Arbeiten aus, die für den Fortgang der Entwicklung auch heute noch zu berücksichtigen sind.

**1976:** Es entstand die erste Empfehlung der Gesellschaft für Informatik (GI) zum Informatikunterricht an allgemein bildenden Schulen. Besonders wichtig ist die Forderung nach einer verpflichtenden Informatik-Bildung für alle Schulabgängerinnen. Die Schwerpunkte dieser Informatikbildung lagen auf algorithmischen Lösungen, auf der Entmystifizierung von Informatiksystemen und deren Einbettung in die Gesellschaft.

„Ein wesentlicher Platz im Informatik-Unterricht kommt einer Projektphase zu, in der das selbständige Entwickeln von Algorithmen in der Form von spielerischem Lernen eingeübt werden sollte.“ [Brauer u. a. 1976]

Mit diesem Vorschlag der Gesellschaft für Informatik wird der Stand der Fachwissenschaft hinsichtlich der inhaltlichen Orientierung des Informatikunterrichts an algorithmischen Strukturen deutlich für eine Umsetzung in der Schule aufbereitet. Die Verflechtung der unterrichtlichen Umsetzung mit Fragestellungen aus dem Anwendungskontext und den Auswirkungen der Informatik auf die Gesellschaft zeigen, dass Informatiker die Verantwortung für die Gestaltung von Problemlösungen als Bestandteil des Problemlöseprozesses berücksichtigt wissen wollen. Dies muss nach Meinung der Autoren dieser Empfehlungen auch im Schulfach Informatik beachtet und damit Gegenstand unterrichtlicher Praxis werden.

Der Vorschlag, eine Projektphase als Bestandteil des Informatikunterrichts vorzusehen, ist bemerkenswert, da er eine methodisch-didaktische Qualität auf der Ebene dieser Empfehlungen erzielt. Für den Erfolg dieser Unterrichtsempfehlungen haben die in den Anhängen beigefügten Beispiele in besonderer Weise beigetragen. Trotzdem blieben noch viele informatikdidaktische Fragen offen, z. B. solche zum Identifizieren von kognitiven Problembereichen einer Einführung von Schülerinnen in die Informatik.

**1977:** Aussagen dazu, worauf eine Lehrperson im Detail zu achten hat, finden sich in den Lehrwerken von Helmut BALZERT ab 1976:

„Um das "Klebenbleiben" an gerätespezifischen Kenntnissen zu Ungunsten des weiteren Verständnisses zu vermeiden, sollte unbedingt zunächst die Problemanalyse, Modellbildung und algorithmische Problemlösung an einer Vielzahl von Beispielen aus unterschiedlichen Problembereichen geübt werden. [...] Es kommt nicht darauf an, daß eine Programmiersprache in kürzester Zeit erlernbar ist, sondern daß sie problemadäquate Sprachstrukturen zur Verfügung stellt, die der menschlichen Denkweise und dem menschlichen Problemlöseverhalten angepaßt sind.“ [Balzert 1977]<sup>3</sup>

„Als didaktisch schwierige Gebiete der Informatik erweisen sich das Variablen-, Feld- und Prozedurkonzept. Auf diese grundlegenden Programmierprinzipien wird daher besonders ausführlich eingegangen.“ [Balzert 1979, S. 8]

Damit wurde eine klare Orientierung an dem informatischen Problemlösungsprozess als didaktische Gestaltung für den Informatikunterricht vorgelegt. Besonders hervorzuheben ist die Beispielauswahl, die nicht aus dem zu diesem Zeitpunkt in den meisten Schulen noch üblichen mathematisch-technischen Aufgabenbereichen stammen. Es werden Fragen der Aufwandsabschätzung (dynamische und statische Komplexität) thematisiert. Auf Fragen des Persönlichkeitsschutzes wird am Rande eingegangen.

Während die vorherigen Arbeiten sehr knapp aus der Bedeutung der aufstrebenden Wissenschaftsdisziplin Informatik auf die Existenz eines gleichnamigen Schulfaches schlossen, führt CLAUS eine umfassendere Diskussion. Dazu er schlägt vor, als Begründungen für das Schulfach Informatik nur solche informatikspezifische Argumente zu akzeptieren, die eine Abgrenzung gegenüber anderen Disziplinen ermöglichen.

„Diese Denkweise in dynamischen Abläufen und im Prozess [...] und das systematische, möglichst fehlervermeidende Entwickeln von Programmen mit Hilfe der Methode der schrittweisen Verfeinerung (strukturiertes Programmieren) sind grundlegend für die neue Wissenschaft "Informatik". [...] Durch diese Denkweise grenzt sich die Informatik zugleich deutlich von der Mathematik ab. Die Mathematik trainiert das Denken in statischen abstrakten Strukturen ("Denken in abstrakten Räumen"). Dies ist jedoch für dynamische Prozesse unzureichend.“ [Claus 1977, S. 22f]

CLAUS geht davon aus und erwartet, dass Informatik als eigenständiges allgemein bildendes Fach in der Schule eingeführt werden wird. Seine Ausführungen stellen ein klares Plädoyer für die schülerorientierte Ausarbeitung und Ausgestaltung konkreter Vorschläge für den Informatikunterricht für alle Schülerinnen dar.

**1979:** Die von GOOS formulierte Qualität der Informatik als konstruktive Wissenschaft eignet sich sehr gut zur weiteren Ausgestaltung der GI-Empfehlungen von 1976 [Brauer u. a. 1976]: „Der Aspekt konstruktive Wissenschaft scheint mir das überzeugendste Argument für die Einführung der Informatik als Schulfach.“ [Goos 1979, S. 2f]

<sup>3</sup>Für die Zitate wurde die 2. Auflage [Balzert 1979, S. 7] herangezogen

**1982:** Erstaunlicherweise dauerte es 17 Jahre, bis die Bedeutung der Informatik bei der Entstehung einer 4. Kulturtechnik [Arlt u. a. 1982, S. 47] einen breiten Konsens unter Bildungsexperten fand: [http://ddi.cs.uni-dortmund.de:8000/ddi\\_bib/gi\\_empfehlung/erfurter\\_resolution.html](http://ddi.cs.uni-dortmund.de:8000/ddi_bib/gi_empfehlung/erfurter_resolution.html). Die Autoren dieses Berichtes setzen sich zum Ziel, für diese neue Kulturtechnik Konzepte zur Umsetzung und Gestaltung zu entwickeln.

**1985:** Während die vorherigen Fachwissenschaftler die Leistungsfähigkeit der Informatik nicht in Frage stellten, setzt sich PARNAS sehr kritisch mit den Grenzen von Software-Engineering-Methoden auseinander.

„Software wird nicht dann zur Benutzung freigegeben, wenn sie nachweisbar korrekt ist, sondern wenn die Häufigkeit, mit der neue Fehler entdeckt werden, auf ein für die Geschäftsleitung akzeptables Niveau gesunken ist. Benutzer lernen, Fehler zu erwarten, und ihnen wird oft erklärt, wie sie bis zur Verbesserung des Programms die Fehler umgehen können.“ [Parnas 1986, S. 59]

Im Folgenden schildert er sehr ausführlich die Vorteile und Grenzen der strukturierten Programmierung, die Verwendung formal spezifizierter abstrakter Schnittstellen zum Verbergen von Information über die Lösung einer Teilaufgabe und die Verwendung kooperierender sequenzieller Prozesse als Hilfsmittel zur Bewältigung der Komplexität. Die Autoren sehen bis zum heutigen Tag ein deutliches Ausbildungsdefizit zur Methodenkritik in der Informatikausbildung. Sie möchten einen Beitrag leisten, dies im allgemein bildenden Informatikunterricht zu berücksichtigen.

**1986:** Mit den GI-Empfehlungen von 1976 [Brauer u. a. 1976] wurden Struktogramme zur statischen Veranschaulichung von dynamischen Abläufen in Informatiksystemen eingesetzt. Darauf gründete die Annahme, dass man für alle schwierigen Bildungsgegenstände der Informatik die passende Veranschaulichung entwickeln könnte. BROOKS widerlegte das mit seiner Analyse der Eigenschaften von Software:

„Invisibility: Software is invisible and unvisualizable. [...] In spite of progress in restricting and simplifying the structures of software, they remain inherently unvisualizable, and thus do not permit the mind to use some of its most powerful conceptual tools. This lack not only impedes the process of design within one mind, it severely hinders communication among minds.“ [Brooks 1986, Brooks 1987]

Die Schlussfolgerung aus dieser Erkenntnis kann nicht sein, dass wissenschaftsorientierter Informatikunterricht in allgemein bildenden Schulen unmöglich ist. Als Analogie sei hier der Welle Teilchen-Dualismus genannt, der kein prinzipielles Hindernis für die Einführung der Physik darstellt. Die Didaktik der Physik hat Lehrkonzepte und kognitive Lehrmodelle entwickelt, die jeweils einen Ausschnitt aus dem Gesamtphänomen erklären, die Komplexität ohne wissenschaftlich Verfälschung reduzieren und sie dem Lehr-Lernprozess zugänglich machen. Diese Aufgabe hat die Didaktik der Informatik für das Phänomen der Software noch zu lösen.

Für den Bereich der Datenbanktechnik begründet LOCKEMANN [Lockemann 1986] fundamentale Ideen – Konsistenz, Konkurrenz und Persistenz, die in der Hochschulbildung enthalten sein sollen. Der Ansatz ist für die Autoren deshalb wichtig, da 1989 beginnend, eine breite Diskussion zur Bedeutung von fundamentalen Ideen der Informatik in der Allgemeinbildung einsetzte, der bis heute die informatikdidaktische Forschung beeinflusst.

**1991:** In den Jahren von 1976 bis 1990 entstand eine Ansammlung von Grundkonzepten der Informatik, die alle im Informatikunterricht allgemein bildender Schulen gelehrt werden sollten. Aus diesen einzelnen „Perlen der Informatik“ entstand aber keineswegs ein lehrmethodisches Schmuckstück: „Ein Potpourri von unzusammenhängenden Grundideen, auch wenn jede treffend vorgestellt wird, empfehle ich keineswegs als Einführung in die Informatik.“ [Nievergelt 1991, 5. Einige klassische Beispiele]<sup>4</sup>

Die Motivation für das hier vorgestellte Forschungsvorhaben gründet genau auf dem bis heute existierenden Defizit. Zwar wurde inzwischen die GI-Empfehlung eines Gesamtkonzeptes der informatischen Bildung von der Primar- bis zur Sekundarstufe veröffentlicht<sup>5</sup>, aber von diesen bildungspolitischen Forderungen bis zur Umsetzung in der Unterrichtspraxis sind mindestens drei Schritte zu realisieren:

1. ein konzeptioneller Rahmen, in dem die Grundideen der Informatik flexibel (Varianten sind möglich) und zugleich logisch verknüpft (Beliebigkeit ist ausgeschlossen) eingebunden werden,
2. qualitativ hochwertige, evaluierte Beispiele und
3. entsprechende Richtlinien und Lehrpläne.

**1992:** Da die Informatik eine junge, sich stürmisch entwickelnde Wissenschaft ist, tritt der Zwang auf, Erkenntnissprünge angemessen in die Lehrdisziplin zu transformieren. Unter dem Titel „Informatik - das neue Paradigma“ wird von BRAUER ein Beitrag vorgelegt, der zu einer grundsätzlichen Neubewertung der Fachstruktur auffordert:

„Als neues Paradigma der Informatik stellt sich ein Computersystem somit dar als Gruppe gleichrangiger, selbständiger, einigermaßen intelligenter Akteure, die bestimmte Aufgaben erledigen und dazu miteinander und mit der Umgebung interagieren.“ [Brauer und Brauer 1992]<sup>6</sup>

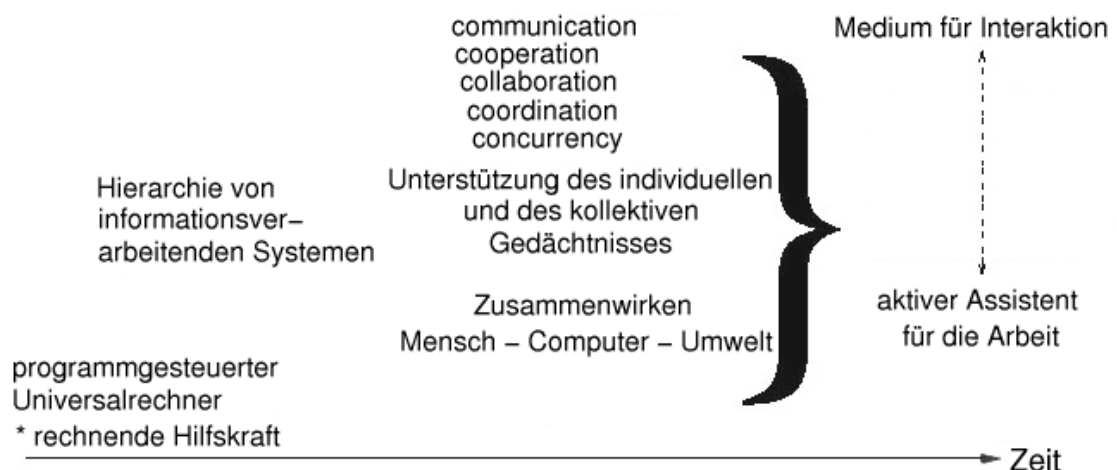


Abbildung 2: Neubewertung der Fachstruktur (nach [Brauer und Brauer 1992])

Eine Schlussfolgerung besteht darin, dass genau diese kooperativen Systeme als Gegenstand und Mittel der Informatikausbildung eingesetzt werden müssen. Stellvertretend sei hier das Beispiel des BSCW

<sup>4</sup>Das Zitat findet sich ebenfalls in [Nievergelt 1993, S. 7].

<sup>5</sup>siehe [Gesellschaft für Informatik 2000]

<sup>6</sup>Für das Zitat wurde [Brauer und Brauer 1995, S. 28] herangezogen.



[Appelt und Busbach 1996] genannt. Wobei die Autoren darauf hinweisen, dass nicht die Auswahl des Systems, sondern dessen Einbindung in den konzeptionellen Unterrichtsrahmen zur Kompetenzentwicklung bei den Schülerinnen beiträgt. Dabei kommt es i. W. auf die didaktische Gestaltung (z. B. Schreibmetapher für gemeinsames Arbeiten) und die Abbildung von Gruppenstrukturen in „virtuelle Räume“ an.

Vielfach dominieren im Informatikunterricht nicht die zentralen Fachkonzepte, sondern Produkte und ihre Bedienung/Benutzung als Missverständnis einer anwendungsorientierten Bildung. Hier kann die Informatik von anderen Wissenschaften mit Ingenieurbezug lernen, die sehr wohl zwischen den technischen Artefakten und den Basiskonzepten, die deren Konstruktion zugrunde liegen in unterschiedliche Bildungsgänge verzweigen.

SHAW fordert: „Let’s organize our courses around ideas rather than around artifacts. This helps make the objectives of the course clear to both students and faculty. Engineering schools don’t teach boiler design – they teach thermodynamics. Yet two of the mainstay software courses – "compiler construction" and "operating systems" – are system-artifact dinosaurs.“ [Shaw 1992]

Eine solche Trennung der Kompetenzbilder bringt Transparenz in die Rolle des Unterrichtsfaches Informatik, in dem die Basiskonzepte der Softwaretechnik zu erlernen sind – ohne Softwareentwickler ausbilden zu wollen.

**1999:** WIRTH: „As teachers, however, we recognize another value in programming: it is in essence the construction of abstractions, the engineering of (abstract) machines. “ [Wirth 1999, S. 3] Im Kontext der Informatik und ihrer Didaktik wird häufig die Frage nach dem Stellenwert der Konkretion von entwickelten Problemlösungen in Form von Programmen gestellt. Gerade auf dem Hintergrund moderner Informatiksysteme mit ihren schier unerschöpflichen Ressourcen wird nach Wirth zu wenig Kraft in die Lehre zur Entwicklung effizienter, ingenieurmässiger Lösungen investiert. Er sieht aber genau hier ein zentrales Handlungsfeld der Informatik.

Eine Konsequenz für die Gestaltung didaktisch nützlicher unterrichtlich umsetzbarer Zielvorgaben besteht nach unserer Überzeugung darin, die konzeptionellen Ergebnisse zum Einen in möglichst verständlicher und klarer Art und Weise umzusetzen, andererseits durchaus grundlegende Überlegungen zur Komplexität von Problemlösungen anzustellen, die am Beispiel unterrichtlich umgesetzt werden.

### 3 Entwicklung der Didaktik der Informatik für Schulen

#### 3.1 Entwicklung in der Bundesrepublik Deutschland

Um für die Bundesrepublik vereinheitlichende Bildungsplanungsmaßnahmen einzuleiten, die den Ländern ein gewisses Maß an Gestaltungsmöglichkeiten eröffnet, entwickelt die BLK (Bund-Länder-Kommission) Rahmenkonzepte. Die KMK (Kultusministerkonferenz) hingegen legt mit ihren Vereinbarungen normierende und damit vergleichbare Abschlussbedingungen für die Sekundarstufe I und das Abitur fest. In der Vereinbarung zur Gestaltung der gymnasialen Oberstufe in der Sekundarstufe II der KMK wird unter dem Punkt Zielsetzung deutlich formuliert, dass in Grundkursen wissenschaftspropädeutische Bildung angestrebt werden muss:

„Die in der gymnasialen Oberstufe zu erwerbenden Kenntnisse, Methoden, Lernstrategien und Einstellungen werden über eine fachlich fundierte, vertiefte allgemeine und wissenschaftspropädeutische Bildung und eine an den Werten des Grundgesetzes und der Länderverfassungen orientierte Erziehung vermittelt, die zur Persönlichkeitsentwicklung und -stärkung, zur Gestaltung des eigenen Lebens in sozialer Verantwortung sowie zur Mitwirkung in der demokratischen Gesellschaft befähigen.“ [Kultusministerkonferenz 1999, S. 4, 5]

Deshalb ist es notwendig, fundierte Studien, was Informatik in der Schule wissenschaftspropädeutisch umfassen muss, vorzulegen. Solche Studien existieren in unzureichendem Maße. Deshalb sind die Autoren dieses Berichts motiviert, einen Beitrag zu leisten.

Die Autoren stützen sich dabei auf allgemeinpädagogische Prinzipien wie Spiralprinzip und Didaktische Reduktion. Der Begriff der Didaktischen Reduktion wird sogar von Lehrpersonen missverstanden, und führt dann zu einer unzulässigen fachlichen Verkürzung. Deshalb sei hier darauf hingewiesen, dass Didaktische Reduktion die Identifizierung der „elementaren Ideen“ eines Sachgegenstandes und die lerngruppenbezogene „Vereinfachung“ bezeichnet, ohne in die Gefahr zu geraten, eine fachlich falsche Vorstellung zu entwickeln (vgl. [Häußler u. a. 1998, S. 201f]). Obwohl das Spiralprinzip allgemein anerkannt ist, kann es in Konflikt geraten mit der Innenstruktur des Lerngegenstandes und mit einer scheinbar zwingenden Fachsystematik. Die Autoren sind sich bewusst, dass es für jede Lerngruppe eine Vielzahl guter Varianten zur Umsetzung eines Spiralcurriculums gibt. Im Bereich der Informatischen Bildung fehlt eine ausreichende Anzahl empirischer Vergleiche, die die Bewertung solcher Varianten erleichtert. Deshalb führen die Autoren exemplarisch empirische Studien durch, die in Kap. 4 zu dem **Modulkonzept** verdichtet werden, das im Anhang exemplarisch konkretisiert wird. Dabei wird im Rahmen dieses Forschungsberichtes ein Ausschnitt der laufenden Forschungsarbeiten präsentiert.

In der Bundesrepublik Deutschland bietet die Fachtagung Informatik und Schule, die seit 1984 meist alle zwei Jahre stattfindet, eine ausgezeichnete Gelegenheit, die Entwicklungsetappen der Didaktik der Informatik für Schulen zu identifizieren. Deshalb wählen die Autoren einige dieser Etappen aus. Damit wird der Versuch unternommen, zusammenfassend auf Umbrüche, Neuorientierung und konzeptionelle Elemente einzugehen. In die Diskussion werden auch wissenschaftliche Veröffentlichungen einbezogen, die nicht auf der Fachtagung erschienen.

**1989:** Im Erscheinungsjahr relativ unbeachtet, aber rückblickend sehr bedeutend ist die Dissertationsschrift von KNÖSS „Fundamentale Ideen der Informatik im Mathematikunterricht“ [Knöß 1989]. Der Forschungsansatz zur Begründung der informatischen Allgemeinbildung aus der mathematischen Bildung im Primarbereich wird von den Autoren sehr kritisch gesehen. Er bildete jedoch die Anregung für weitergehende Forschungsarbeiten (z. B. [Schwill 1993]). Knöß meint, dass der Mathematikunterricht in der Primarstufe die Darstellung, Realisierung und Qualität von Algorithmen und Datenstrukturen den Lernenden nahebringt. Die Verkürzung fundamentaler Ideen der Informatik auf Algorithmen und Datenstrukturen bleibt hinter den Bildungserfordernissen der Schulinformatik in dieser Zeit zurück.

Relativ unerkannt, hatte sich ein Spannungsfeld aufgebaut. Lehrpersonen gestalteten ihren Unterricht nach den Empfehlungen der GI von 1976<sup>7</sup>, indem sie am Beispiel einer prozeduralen Programmiersprache Algorithmen und Datenstrukturen thematisierten. Das sah nach einem erfolgreichen Informatikunterricht aus. Gleichzeitig verloren die befehlsorientierten Unterrichtsbeispiele immer mehr an Motivationskraft im Vergleich zu Informatikprodukten, die Schülerinnen im Alltag begegneten. Deshalb wird der Vortrag von PESCHKE im nachhinein als bedeutender Versuch einer Neuorientierung der informatischen Bildung anerkannt, der aber 1989 heftig umstritten war.

„Probleme wie vernetzte Gesellschaft und informationelle Selbstbestimmung, die automatisierte Fabrik und soziale Gestaltbarkeit, fehlerhafte informationstechnische Systeme und Verantwortlichkeit können über den algorithmischen Zugang nicht befriedigend behandelt werden.“ [Peschke 1989, S. 91ff]

Damit geht PESCHKE in korrekter Weise auf den wissenschaftstheoretischen Stand der Informatik ein (vgl. Kap. 2). Da er aber massiv gegen die Stärkung des Unterrichtsfaches Informatik argumentiert, treten aus heutiger Sicht sehr ungünstige Lösungsversuche ein. So wurde in der Folge z. B. einer Pseudo-Integration informatischer Fachinhalte über Projekte, die an ein Leitfach gebunden wurden, propagiert und führten zu einer deutlichen Verschlechterung der Stellung und der Notwendigkeit des Schulfachs Informatik. Die Didaktik der Informatik war zu diesem Zeitpunkt noch kein etabliertes Teilgebiet der Informatik. Es wurden aber für die wenigen Standorte der Lehrerbildung Informatik bereits Lehrveranstaltungen angeboten, die von Forscherinnen des computerunterstützten Unterrichts (CUU) getragen wurden. Einer engagierten Lehrerschaft und der Fachzeitschrift LOG IN, die kontinuierlich gute Beispiele für den Informatikunterricht veröffentlichte, gelang es, den Fortbestand der Schulinformatik zu sichern. Die Didaktik der Informatik existierte also in einer Vielzahl dokumentierter Unterrichtserfahrungen. Es fehlten ihr zu diesem Zeitpunkt tragfähige Konzepte, die dem berechtigten Anspruch von PESCHKE gerecht wurden.

**1991:** SCHUBERT fordert die Einbeziehung von theoretischen Elementen in das Schulfach Informatik. Sie macht deutlich, dass der Bereich Nichtdeterminismus der Wissenschaft Informatik bislang unterrichtlich kaum berücksichtigt wird, obwohl ihm für die Weiterentwicklung der Informatik in der Schule eine wichtige Rolle zufallen kann. Des weiteren gibt sie an, dass für das Schulfach Informatik geeignete Systemumgebungen notwendig sind, die als Basis für die Erarbeitung inhaltlicher Fragen nützlich sind. Interessant ist hierbei der Vergleich mit der im gleichen Jahr veröffentlichten Stellungnahme von Nie-

<sup>7</sup> siehe [Brauer u. a. 1976]

vergelt, der als Fachwissenschaftler argumentierend ähnliche Schlussfolgerungen zieht (vgl. Kap. 2.2, S. 9, 1991).

Sowohl NIEVERGELT wie auch SCHUBERT begründen Vorgehensweisen, die besonders für die informatikdidaktische Ausbildung von Lehramtsstudierenden angewendet werden können. Damit wird die Didaktik der Informatik für Schulen als verbindendes Element von Hochschule und Unterrichtspraxis stärker sichtbar.

**1993:** Aus der Analyse der Bereiche der Kerninformatik [Schwill 1993, S. 25ff] leitet SCHWILL drei fundamentale Ideen der Informatik für die allgemeine Bildung ab:

- Algorithmisierung,
- strukturierte Zerlegung und
- Sprache.

In Baumstrukturen werden diesen drei fundamentalen Ideen mit weiteren fundamentalen Ideen unterstellt. Es ist nicht nachzuvollziehen, warum SCHWILL gerade auf 55 fundamentale Ideen kommt. Die Vermutung, dass der subjektive Erfahrungshintergrund des Forschers die Begründungsintensität stark beeinflusste, liegt nahe. In den Studien findet sich keine Diskussion von Ideen, die zwar geprüft, dann aber verworfen wurden. Der weitergehende Anspruch einer „vollständigen Kollektion aller fundamentalen Ideen der Wissenschaft“ [Schwill 1993, S. 23] wird mit einer theoretischen Methodik verbunden, die höchst fragwürdig erscheint. Es soll „von den Inhalten einer Wissenschaft zu ihren Ideen abstrahiert“ [Schwill 1993, S. 23] werden, ohne zu beachten, dass dieser Auswahlprozess einen historischen Kontext besitzt; also höchstens eine Momentaufnahme der Ideenkollektion zu einem konkreten Zeitpunkt liefern könnte.

Die Autoren sehen in der fundamentalen Idee „Teamarbeit“, die der strukturierten Zerlegung und dort dem Zweig der Modularisierung zugeordnet wurde, ein Problem und eine Chance dieses theoretischen Ansatzes zugleich. Die Chance liegt darin, dass die fundamentalen Ideen der nicht formalisierbaren Bereiche der Informatik in das Bildungskonzept einbezogen werden können. Das Problem besteht darin, dass die partizipative Softwareentwicklung mit starker Betonung der Kommunikationsprozesse nicht das letzte Blatt eines Baums der fundamentalen Ideen sein kann, denn Softwareentwicklung ist ein kooperativer Prozess zwischen Menschen, der sich nicht allein mit Algorithmisierung und strukturierter Zerlegung erklären lässt.

Dieser – aus Sicht der Autoren – besonders bedeutsame Beitrag zur Fundierung der Didaktik der Informatik führte zu einer Polarisierung in der Lehrerschaft. Die Kritik ist nachvollziehbar, da theoretische Informatik in den fundamentalen Ideen überproportional häufig vertreten ist, technische Informatik fast fehlt. Schwerwiegender erscheint den Autoren der fehlende Zugang zu soziotechnischen Systemen und den damit verbundenen Anwendungsbereichen.

Die Stärke der Konzepts fundamentaler Ideen für das Schulfach Informatik besteht in der fachlichen Absicherung von Inhalten. Informatik in der Schule kann aber nicht allein über die Fachsystematik begründet werden, sondern benötigt den allgemeinpädagogischen Kontext. Das heißt, Inhalte sollten sich aus der Lebenswelt der Schülerinnen ergeben und müssen als epochale Schlüsselfragen [Klafki 1985] im Informatikzusammenhang thematisiert werden.

BAUMANN legt ein Konzept zur Ausgestaltung des Unterrichts für die Sekundarstufe II vor, in dem er die „zeitgemäße didaktische Weiterentwicklung des Informatikunterrichts“ [Baumann 1993, S. 12] in Form eines systemorientierten Ansatzes fordert: „Damit ist informatisches Problemlösen als Systementwicklung (Entwicklung eines Informatiksystems) charakterisiert, wobei [...] die [...] Informatikmethoden ihre zentrale Stellung behalten. Ferner wird der Computer nicht als isoliertes Einzelgerät, sondern als Teil umfassender Systeme gesehen.“ Der Ansatz wird von BAUMANN „als Synthese der bisherigen Ansätze (einschließlich der Vorgeschichte)“ charakterisiert. [Baumann 1993, Zitatenkollage, S. 12]

Ausgehend von drei Leitfragen, entwickelt er jeweils ein fachliches Richtziel:

- A) „Wie werden Informatiksysteme entworfen, programmiert und damit zum Lösen lebensweltlicher Probleme befähigt?  
Problemlösen als methodischer Entwurf von Informatiksystemen.
- B) Wie sind Informatiksysteme aufgebaut, wie wirken ihre Komponenten zusammen und wie ordnen sie sich in umfassendere soziotechnische Systemzusammenhänge ein?  
Struktur und Funktion von Informatiksystemen im soziotechnischen Kontext.
- C) Wo liegen die prinzipiellen Grenzen technischer Informationsverarbeitung, und was ist unter Information und Kommunikation überhaupt zu verstehen?  
Prinzipielle Grenzen technischer Informationsverarbeitung sowie grundlegende Konzepte von Information und Kommunikation.“ [Baumann 1993, S. 12f]

BAUMANN liefert damit auf dem Hintergrund eines zeitgemäßen Bildungsbegriffs (nach Klafki) die Legitimation für Informatik als eigenständiges Schulfach des Sekundarbereichs II. In der Ausformung der Zieldimensionen orientiert er sich sehr klar an dem Nutzen theoretischer Begriffe als konstitutives Element des Bildungsgehalts des Faches (vgl. [Baumann 1993, z. B. S. 13]).

Die GI legt „Empfehlungen zur Gestaltung des Informatikunterrichts für die Sekundarstufe II“ vor. Damit wird eine Tradition fortgesetzt, die 1976 mit der Veröffentlichung der „Zielsetzungen und Lerninhalte des Informatikunterrichts“ [Brauer u. a. 1976] einen Meilenstein für die Entwicklung der Informatik in der Schule setzte. Es wird herausgestellt, dass „Sichtweisen benötigt [werden], die die Entwicklung der Informatik aufnehmen und den [...] Anforderungen der Allgemeinbildung genügen.“ [Schulz-Zander u. a. 1993, S. 206]

Im Weiteren werden Ziele und Inhalte des Informatikunterrichts angegeben. Die vorgeschlagene Struktur umfasst drei Sichtweisen. Im Folgenden werden die für den thematischen Zusammenhang zentralen Elemente der Gliederung aufgeführt.

- A) **Mensch-Computer** mit den Elementen: Exemplarische Methoden und Verfahren der Modellierung, Möglichkeiten der Gestaltung von Benutzerschnittstellen und Software-Entwicklungsumgebungen, Möglichkeiten von Teamorganisation und Selbstorganisation.
- B) **Formalisierung und Automatisierung geistiger Arbeit** mit Elementen zur theoretischen Fundierung der Informatik: einfache Automatenmodelle und zugehörige Grammatiken, Wirkprinzipien und die Architektur von Informatiksystemen finden Erwähnung.

**C) Informatiksysteme, Gesellschaft und Umwelt** richtet das Augenmerk auf fundierte Einsichten in Chancen, Risiken und Grenzen eines verantwortbaren Einsatzes von Informatiksystemen, das Grundrecht auf informationelle Selbstbestimmung, thematisiert die Beurteilungsfähigkeit von Auswirkungen der Anwendungen der Informatiksysteme auf die Lebens- und Arbeitswelt, auf Politik und Umwelt sowie die kulturelle, historische und anthropologische Dimension des Einsatzes von Informatiksystemen.

(vgl. [Schulz-Zander u. a. 1993, S. 208ff])

Den beiden eher theoretisch orientierten Veröffentlichungen von SCHWILL und BAUMANN stehen die notwendigerweise pragmatisch orientierten Empfehlungen der GI von 1993 gegenüber. Die Stärken dieser Empfehlungen bestehen darin, dass gegenüber vorherigen Empfehlungen deutlich gemacht wird, dass „eine exemplarische Behandlung von Themen im Unterricht [vorzusehen ist], die alle Sichtweisen abdeckt. [...] Bei der Gestaltung des Unterrichts ist anzustreben, die verschiedenen Sichtweisen zu integrieren.“ [Schulz-Zander u. a. 1993, S. 208]

Damit wird der Zergliederung der Inhalte und der Reduzierung auf Einzelaspekte eine Absage erteilt. Überdauernde Grundkonzepte (= Bewährtes) werden benannt, z. B. „Entwicklung von Algorithmen und Datenstrukturen, [...] einfache Automatenmodelle und zugehörige Grammatiken“, es wird deutlich auf verschiedene Paradigmen hingewiesen, und gefordert, dass sie ihren Niederschlag im Unterricht finden sollen.

Die vorgelegten Empfehlungen machen deutlich, dass die GI die kontinuierliche Weiterentwicklung der Informatik in der Schule befürwortet.

**1995:** FRIEDRICH hält die Diskussion über den allgemeinbildenden Charakter der Informatik für abgeschlossen und fokussiert auf folgende Grundprinzipien:

- Grundprinzipien des Aufbaus, des Funktionierens und der Wirkung von Maschinen zur Verarbeitung von Informationen;
- Strategien der Problemlösungen mit Werkzeugen der Informatik, Betrachtungen zu deren Möglichkeiten und Grenzen;
- Gesichtspunkte und Konsequenzen der Abstraktion und Modellbildung mit Mitteln der Informatik.“ [Friedrich 1995, S. 38]

Er fordert die Fachdidaktik auf, die Ausgestaltung des Schulfachs Informatik im Ensemble der anderen Fächer in der Schule – sowohl bezogen auf die Inhalte (hier vor allem Begriffsbildung), wie auch auf die spezifischen informatischen Arbeitsmethoden – zu konturieren.

Bei den Anforderungen wird allerdings z. B. nicht spezifiziert, welche Modellbildung der Informatik für die Schule gewählt werden soll.

Für ENGBRING stellt der Begriff „Technik“ die unumstrittene Gemeinsamkeit aller Inhalts- und Begründungsfragmente dar: „Informatik kann als Teil technischer Bildung einen Beitrag zur Erklärung technischer Phänomene sowie zur Einschätzung, Beurteilung und Bewertung von Technik leisten, so wie die Naturwissenschaften in den Schulen einen Beitrag zur Erklärung von Phänomenen der Natur leisten. Dieser letztgenannte Aspekt technischer Bildung ist in der gegenwärtigen Situation für das

gesellschaftliche Zusammenleben und die Entmystifizierung von Technik, speziell von Computern, genauso wichtig oder gar wichtiger, als es heute die Naturwissenschaften für die Entmystifizierung von Naturphänomenen noch sind.“ [Engbring 1995, S. 76].

Die Chance besteht darin, dass mit dem Schulfach Informatik in der Schule der Themenbereich „Technik“ und „Gestaltung der Technik“ einen den gesellschaftlichen Anforderungen angemessenen Raum erhält. Andererseits kann in einem neu aufzurichtenden allgemein bildenden Fach nicht bereits zu Beginn erwartet werden, dass ein allgemeiner Konsens über die Basis, die das Schulfach konstituiert, erreicht wurde.

SCHUBERT und SCHWILL dokumentieren Ergebnisse einer fachdidaktischen Arbeitsgruppe zur „Strukturierung der Schulfachs Informatik aus der Sicht der Fachwissenschaft“. Die Arbeitsgruppe stellt fest, dass das Konzept der fundamentalen Ideen allein nicht ausreicht, um zu einer Systematik zu gelangen, weil wissenschaftstheoretische Sichtweisen der Informatik (Woher kommt, was ist, wohin geht die Informatik?) und Fragen nach dem Beitrag des Schulfachs Informatik zur Allgemeinbildung nur unzureichend berücksichtigt werden. (vgl. [Schubert und Schwill 1996]).

**1997:** KRÄMER stellt mit ihren Thesen über den Umgang mit dem technischen Artefakt Computer die Zeitgebundenheit der Inhalte und den damit verbundenen Perspektivenwechsel heraus: „Damit wird, was als eine Denkleistung zählt, abhängig von den sich historisch wandelnden kulturellen Praktiken unseres Zeichengebrauchs. [...] Was der Computer leistet, hat [...] mit der Automatisierung von Symboloperationen, die wir immer schon extern vollzogen haben [zu tun]. Der Computer zeigt sich [...] als der apparative Vollzug von Kulturtechniken, die auf dem Einsatz von „symbolischen Maschinen“ beruhen“ [Krämer 1997, S. 8]. Sie fügt dem „kanonischen“ dualen Schema: „[der] Unterscheidung von instrumentellem und kommunikativem Handeln“<sup>8</sup> eine Überlegung für die Möglichkeit einer dritte Modalität unseres Handelns hinzu: die „spielerische Interaktion“ (vgl. [Krämer 1997, S. 12f]). Daraus ergibt für sich abzeichnende Veränderungen die Fragestellung: „[Kann die instrumentelle Perspektive] erschließen, worin dessen [gemeint ist die Nutzung des Computers, die Autoren] zukunftsträchtiges Potential liegen kann?“ [Krämer 1997, S. 7]

Diese Frage kann nach Meinung der Autoren am Ende des Artikels allgemeiner formuliert werden, indem sie um die beiden von KRÄMER zusätzlich herangezogenen Perspektiven Denkzeug, Spielzeug erweitert wird.

Für den hier betrachteten Zusammenhang ist es wichtig, festzuhalten, dass die drei verschiedenen Perspektiven zwar historisch in Folge herausgearbeitet wurden, aber keine als ausschließlich gültige Perspektive angesehen werden kann. Jede der Perspektiven hat ihren Gültigkeitsbereich und läßt unterschiedliche Entwicklungsmöglichkeiten erkennen.

HUBWIESER und BROY orientieren in ihrem Beitrag auf die primäre Zielstellung eines Informatikunterrichts, der „den Schülern wichtige Fähigkeiten zum Umgang mit Informationen [vermittelt].“ Dies

<sup>8</sup> „[als] Differenz zwischen einem Tun, das erfolgsorientiert ist und auf die technikgestützte Leistungssteigerung im Umgang mit Sachen abzielt und einem Tun, das verständigungsorientiert ist und die sprachvermittelte Anerkennung anderer Personen impliziert.“ [Krämer 1997, S. 12]

wollen sie erreichen, indem sie „geeignete mentale Modelle anbieten, [...] die charakteristischen Eigenschaften von Informations- und Kommunikationssystemen [...] zu begreifen und [...] sie zu beherrschen.“ [Hubwieser und Broy 1997, S. 41] Die Modellierung wird als Unterrichtsprinzip in das Zentrum der konkreten Arbeit der Informatik als Schulfach gerückt. Im Zusammenhang mit methodischen Vorschlägen wird deutlich herauskristallisiert, dass als Ausgangspunkt eines so orientierten Unterrichts Problemstellungen aus der Praxis zur Problemgewinnung gewählt werden sollten, die – in projektorientierte Unterrichtssequenzen heruntergebrochen – die Phasierung des Modellbildungs- und Simulationsprozesses im Unterrichtsprozess widerspiegelt (vgl. [Hubwieser und Broy 1997, S. 42f]).

HUBWIESER beschreibt zusammenfassend die Phasierung einer so verstandenen Unterrichtsgestaltung:

- Problemgewinnung,
- informelle Problembeschreibung,
- Formale Modellierung,
- Realisierung von Lösungsansätzen/Simulation und
- Bewertung (vgl. [Hubwieser 1997, S. 216]).

Mit diesem Beitrag wird eine theoriegeleitete Klammer zum Schulfach Informatik vorgelegt. Der Ansatz wird auch als informationszentrierter Ansatz bezeichnet (vgl. [Hubwieser und Broy 1996]). Durch diesen Vorschlag wird die Modellierung im Informatikunterricht in das Zentrum der didaktischen Überlegungen gerückt. Allerdings wird die Realisierung nicht „unter den Tisch gekehrt“, sondern stellt, wie in der Phasierung durch HUBWIESER deutlich wird, einen notwendigen Schritt der Unterrichtsgestaltung dar.

WITTEN und PENON dokumentieren und extrapolieren Erfahrungen, die sie im Zusammenhang mit der Nutzung elaborierter Informatiksysteme im Informatikunterricht und als Grundlage zur schulweiten Präsentation von Ergebnissen gemacht haben. Dabei stellen sie ihre Unterrichtserfahrungen mit vernetzten Systemen als Unterrichtsgegenstand vor. In der Reflexion eines durchgeführten eigenständigen Kurses „Telekommunikation“ weisen sie auf Perspektiven für den Informatikunterricht hin: „Ein nächster Schritt wäre es, Teile des Informatik-Unterrichts, z. B. Softwareprojekte, mit Hilfe von Workgroup-Software zu realisieren. Hier bietet sich das für Schulen frei erhältliche Programm BSCW der GMD an. Dieses Programm ermöglicht für einen geschlossenen Benutzerkreis den gezielten Austausch von Dokumenten und deren Verwaltung, eine fortlaufende ergebnisorientierte Problemdiskussion und vieles mehr.“ [Witten und Penon 1997, S. 173] Außerdem betonen sie, dass Fragen der Codierung, der Computersicherheit und der Kryptologie im Zusammenhang mit dieser Unterrichtsgestaltung eine neue Bedeutung erlangen, die auch geschichtlich interessant sei (vgl. [Witten und Penon 1997, S. 173]).

Die Darstellung macht deutlich, dass die Orientierung des Informatikunterrichts an aktuellen Problemstellungen dazu führen kann, dass bereits „überholt“ geglaubte Fragestellungen der Informatik in der Schule unter einer neuen Perspektive Bedeutung erlangen. Auch Bereiche, die bereits seit langem für wichtig erachtet werden, aber der unterrichtlichen Umsetzung nur schwer zugänglich sind, sollten (zumindest versuchsweise) umgesetzt werden, damit Erfahrungen mit der didaktischen Gestaltung informatischer Inhalte gesammelt werden können, denen ggf. zukünftig eine wichtige Rolle zukommt.



BERGER dokumentiert die Untersuchung des Computer-Weltbildes von Informatiklehrerinnen an deutschen Schulen. Die in der Studie angewandte Forschungsmethodik basiert auf Verfahren der aktuellen qualitativen Sozialforschung. Die dokumentierten Ergebnisse der Untersuchungen werden im Folgenden zusammengefasst.

„Insbesondere bei den akademisch ausgebildeten Informatiklehrern ist [...] eine Hinwendung zu dem als fundamental eingeschätzten Konzept der Algorithmik [festzustellen]. [...] Sie] neigen [...] zu einer Betonung des Grundlegenden und Bleibenden [...]. Das Lernen lokaler Taktiken [...] tritt zurück zugunsten des Erwerbs globaler Strategien (Denken in Prozessen und Systemen)“ [Berger 1997, S. 30]. Zum Themenfeld 'Informatik als Wissenschaft' führt BERGER aus: „Standardisiert man die individuellen Beschreibungen, so läßt sich die überwiegende Mehrheit [...] den vier Hauptkategorien Informationswissenschaft, Strukturwissenschaft [...], Computerwissenschaft und Algorithmentheorie zuordnen“ [Berger 1997, S. 29f].

Ergebnisse zu dem ebenfalls untersuchten Themenbereich „Bild vom Lehren und Lernen von Informatik“ [Berger 1997, S. 33ff] stellt BERGER folgendermassen dar: „Das traditionelle Paradigma Schule – charakterisiert durch Schlüsselbegriffe wie Unterricht, Hausaufgabe, Klassenarbeit, lehren, erziehen, prüfen, benoten etc. – wird wenn auch nicht geradezu verdrängt, so doch zunehmend ergänzt und überlagert von einem neuen Paradigma Berufswelt mit den Leitkonzepten Projekt, Produkt, Team, Diskussion, beraten, delegieren, mitbestimmen und kooperieren[...] Allerdings ist dieser Wechsel zumeist nicht als Folge bewußter didaktischer Innovation und zielgerichteter Entwicklung eines neuen Unterrichtsstils zu erklären. [...] Prononciert könnte man formulieren: Im innovativen Schulfach Informatik findet Innovation zur Zeit weniger von innen statt, durch den innovativen Lehrer, der ein neues Paradigma des Lehrens und Lernens findet – als vielmehr von außen, durch ein neues Paradigma, das 'seinen Lehrer findet' und ihn, auch den durchaus traditionell eingestellten, zunehmend zu innovativen Mustern greifen läßt.“ [Berger 1997, S. 38]

Bezogen auf diese Ergebnisse wird deutlich, dass die Veränderung der Gewichtung der Kernbereiche und der zentralen Konzepte des Informatikunterrichts über die Zeit eine zentrale Größe in der Selbsteinschätzung der Informatiklehrerinnen darstellt. Nicht unterschätzt werden sollten die Aussagen bzgl. der Unterrichtskultur und damit der didaktisch-methodischen Orientierung des Informatikunterrichts. Für die Didaktik der Informatik bedeutsam erscheint uns die Feststellung eines Paradigmenwechsels hin zu modernen Leitkonzepten, die mit innovativen Mustern unterrichtlich umgesetzt werden. Diese Ergebnisse sollten produktiv in die Didaktik der Informatik einfließen, belegen sie doch in eindrucksvoller Weise, dass dem Schulfach Informatik eine zentrale Rolle in der Umsetzung notwendiger innovativer Konzepte für die Schule zukommt. Damit kann das Schulfach Informatik zum Leitfach für notwendige Umgestaltungen der schulischen Praxis werden.

**1999:** MAGENHEIM, SCHULTE und HAMPEL dokumentieren einen fachdidaktischen Ansatz zur objektorientierten Modellierung unter Bezugnahme auf die Konzepte Dekonstruktion (als Möglichkeit zur Durchdringung) und Konstruktion (als Erweiterung) existierender Informatiksysteme. Damit soll „Schülerinnen und Schülern die Einsicht in Verfahren der objektorientierten Modellierung sowie in relevante Methoden der Gestaltung und Bewertung von Informatiksystemen gegeben werden“ [Magenheim u. a. 1999, S. 149]. Als Bezug wird die systemorientierte Didaktik der Informatik ausgewiesen. Projektorientierter

Unterricht wird für notwendig erachtet, und in der Konkretion und Ausgestaltung die objektorientierte Modellierung favorisiert.

Das Konzept lässt zu diesem Zeitpunkt allerdings Fragen nach der konkreten Umsetzung unbeantwortet. Es ist deutlich, dass der Arbeitsgruppe MAGENHEIM, SCHULTE u. a. eine nicht zu unterschätzende Arbeit bevorsteht. Ein zentrales Problem besteht darin, eine notwendige Komplexität der zu dekonstruierenden Systeme mit einer Erweiterungsmöglichkeit so zu verbinden, dass sie handhabbar und offen bleiben. Ein weiteres Problem stellt die ausschliessliche Orientierung des Ansatzes auf die Softwaretechnik dar. So werden die Elemente der technischen Informatik und damit der gesamte Bereich der Wirkprinzipien bei dem Konzept nicht in den Blick genommen.

HUMBERT unternimmt in seinem Beitrag den Versuch, eine Bestandsaufnahme des Faches Informatik im allgemeinen und in der Schule, speziell der gymnasialen Oberstufe, zu leisten. Sie soll dem langfristigen Ziel dienen, einen Beitrag zur Entwicklung eines verpflichtenden und aufeinander abgestimmten Curriculums für den Informatikunterricht in den Sekundarstufen I und II zu leisten. Es wird ein Überblick über didaktische Zugänge zur Erarbeitung informatischer Inhalte im Unterricht gegeben und eine Beziehung zu fachwissenschaftlichen Sichtweisen der jeweiligen Thematik hergestellt. Als mögliche zentrale Gegenstände einer Theorie der Informatik werden Wirkprinzipien und die Verantwortung der Informatiker, der technische Umgang mit Wissen bzw. Information, Sprachen zur Codierung von Entwürfen, der Computer als symbolverarbeitende Maschine sowie Sinn und Zweck der Informatik vorgestellt. Der Autor umreißt mit dieser komprimierten Beschreibung eines komplexen Sachverhalts den Diskussionskontext, innerhalb dessen sich eine curriculare Fundierung der Informatik in der Schule vollziehen sollte. HUMBERT klassifiziert die vorgestellten Beispiele nach den Kategorien Zugang, Modellierungsmethode und Sprache. Hinsichtlich des Zugangs ordnet er die vorgestellten Beispiele den didaktischen Konzepten eines anwendungs- bzw. gesellschaftsorientierten Ansatzes zu. Es werden in den angedeuteten Beispielen imperative, objektorientierte und logische Modellierungsmethoden verwendet. Der Autor präsentiert Vorüberlegungen zu einem Spiralcurriculum des Informatikunterrichts, der bereits in der Sekundarstufe I zu einer fundierten informatischen Grundausbildung führen sollte, die die persönliche Erstbegegnung der Schülerinnen mit dem Informatikunterricht positiv prägen soll. Als Leitlinien eines solchen Spiralcurriculums werden u. a. benannt: Informatische Grundlagen der Nutzung informationstechnisch unterstützter Gruppenarbeit, Ablösung der strukturierten Programmierung durch die objektorientierte Modellierung, Einsatz alternativer Problemlösungsmethoden der Informatik zur Förderung von Denkstrukturen, die nicht nur zur Bearbeitung von informatikspezifischen Problemstellungen genutzt werden (vgl. [Humbert 1999]).

Die Skizze eines 'Curriculumvorschlags' für die Jahrgangsstufen 5 – 10 und ergänzende Varianten sind in ihrer Darstellung jedoch so knapp, dass damit mehr Fragen erzeugt als Antworten gegeben werden.

### 3.2 Internationale Diskussion

**1993:** ACM (Association for Computing Machinery) Model High School Computer Science Curriculum<sup>9</sup>:

In der aus sieben Bereichen<sup>10</sup> bestehenden Themenübersicht des Curriculums, fällt der Bereich Social, Ethical and Professional Context ins Auge. Die quantitative Ausgestaltung dieses mit den Inhalten: ethische, soziale, juristische und kulturelle Aspekte, sowie Verständnis der historischen Entwicklung der Informatik auszugestaltenden Bereichs ist recht gering, stehen doch nur 11 – von insgesamt 303 – Stunden – dafür zur Verfügung. In der Einführung wird deutlich gemacht, dass die Beschäftigung mit Details vermieden werden muss: „Because the details of the technology change from day to day, keeping up with those details is difficult and often unproductive. Therefore the study of the subject must concentrate on the fundamental scientific principles and concepts of the field. [...] The focus of the course is on fundamental concepts of computer science. Several model course curricula show how different settings can be used to illustrate these concepts. As much as possible, students will conduct experiments and write programs that demonstrate the abstract concepts, confirm the theory and demonstrate the power of computers“ [Association for Computing Machinery 1997, Introduction, Motivation]. Verschiedene Methoden zur Erarbeitung werden – auch in Kombination – für die Bereiche für möglich erachtet: die Erarbeitung mit Hilfe der Programmierung von Beispielen, durch Übungen, in Projekten und mit Hilfe von Berichten für die zentralen Bereiche – aber auch die Arbeit mit Anwendungspaketen. In der Zusammenfassung wird deutlich formuliert, dass es sich um ein verpflichtendes Curriculum für alle Schülerinnen handelt, dessen Fundamente zu vermitteln sind. Diese liegen – nach dem o. g. Rahmen – in den Bereichen [Rechner-] Architektur, Betriebssysteme, Algorithmen und Datenstrukturen, Programmiersprachen und Softwaretechnik (in Summe über 85%, wenn die Einführung in eine konkrete Programmiersprache hinzugenommen wird). „Computer science is essential for the education of every citizen. The ACM model high school curriculum identifies the essential concepts in computer science which every high school student must understand“ [Association for Computing Machinery 1997, Summary].

Die curriculare Arbeit der ACM ist von grosser Kontinuität geprägt. Damit verläuft allerdings die Weiterentwicklung in kleinen Schritten. Die Aufnahme des neuen Bereichs Social, Ethical and Professional Context ist quantitativ sehr gering. Insgesamt ist festzustellen, dass die curriculare Orientierung an gewachsenen Teilgebieten der Informatik erfolgt. Eine konzeptionelle Orientierung als Klammer des Curriculums wird nicht expliziert.

**1994:** IFIP/UNESCO (International Federation for Information Processing/United Nations Educational, Scientific and Cultural Organization): Informatics for secondary education: a curriculum for schools

Durchgängig wird der Begriff Informatics benutzt: „The following definition[s] have been used by the working party: Informatics: the science dealing with the design, realisation, evaluation, use and maintenance of information processing systems; including hardware, software organisational and human aspects, and the industrial, commercial, governmental and political implications (UNESCO/IBI)“ [van Weert u. a. 1994, S. 8].

<sup>9</sup>[Association for Computing Machinery 1993] und [Association for Computing Machinery 1997], die 97er Fassung wurde gegenüber 1993 nur unwesentlich verändert.

<sup>10</sup> Algorithms; Programming Languages; Operating Systems and User Support; Computer Architectures; Social, Ethical and Professional Context; Computer Applications; Additional Topics

Als Hauptzielbereiche des Curriculums werden die folgenden Elemente ausgewiesen: „1. Computer Literacy [...], 2. Application of IT Tools in other Subject Areas [...], 3. Application of Informatics in other Subject Areas – Students should be able to use methods and techniques from informatics in combination with information technology tools to solve problems in other subject areas. This main objective should preferably be addressed in General Education at the Advanced Level., 4. Application of Informatics in Professional Areas – Students should be able to use methods and techniques from informatics in combination with information technology tools to solve professional problems from business and industry. This main objective should preferably be addressed in Vocational Education at the Advanced Level

[...] General Education at the Advanced Level – Having fully met Objective 1 (Computer Literacy), the focus is on Main Objectives 2 and 3 [...] Objective 3 involves the following sequence of problem solving skills using the techniques and tools of informatics, namely the methodical modelling of the problem, design of an algorithmic solution, programming the solution either in a general or computer specific way, and an evaluation of the proposed solution. This implies that students have developed a functional model of a computer system and its programming environment.“ [van Weert u. a. 1994, S. 9f]

Trotz der im Titel deutlich ausgewiesenen Orientierung auf Informatik wird in den Zielbereichen 1 und 2 sehr stark auf eine an Anwendungen orientierte Ausrichtung expliziert. Die Informatik ist diesem curricularem Entwurf zufolge nachgelagert. Dies soll auf der anderen Seite durch eine grundlegend am problemlösenden Denken orientierten didaktischen Grundhaltung untersetzt werden. Dabei wird – nach unserer Einschätzung – aus den vorgeschlagenen Bereichen kein durchgängiges Informatikcurriculum, sondern ein Konzept, das die Informatik isoliert und künstlich von IT zu trennen versucht.

Die Vorschläge ACM und IFIP/UNESCO lassen unterschiedliche Ansätze erkennen, die der Informatikdefinition zuzuschreiben ist, die jeweils zur Anwendung kommt. Die Ergebnisse auf der curricularen Ebenen weichen hinsichtlich des fachlichen und des pädagogischen Hintergrunds voneinander ab: IFIP/UNESCO fordert Problemlösekompetenz, legt Wert auf Modellierung und erklärt die Informatik als Hilfsmittel zur Problemlösung. ACM hingegen legt ein „konservatives“ Informatikverständnis zugrunde und fordert, dass die Inhalte auf zentrale wissenschaftliche Prinzipien und Konzepte orientiert zu erarbeiten sind. Gerade aus dieser Spannung heraus ist eine verstärkte internationale Auseinandersetzung um Informatik als Inhalt allgemeiner Bildung notwendig. Es wäre zu begrüßen, wenn die beiden curricularen Vorschläge so miteinander verzahnt würden, dass die Fachkonzepte der Informatik auf der Folie moderner pädagogischer Konzepte erfolgreich vermittelt werden.

### **1996: EBERLE: Überlegungen zu einer Didaktik der Informatik**

In einer „Abschliessenden Übersicht“ fasst EBERLE die im Zusammenhang mit seiner Habilitationsschrift [Eberle 1996] entwickelten Elemente einer „Didaktik der Informatik ...“ zusammen. Dabei macht er u. a. deutlich:

„Das Schulfach Informatik [...] hat sich seit seinen Anfängen aus einer Rechnerkunde zu einem Bildungsgefäß entwickelt, über dessen Inhalte stark auseinanderdriftende Ansichten bestehen. Je nach Phase, Ansatz oder pragmatischer Festschreibung wurden und werden Themen der Wissenschaftsdisziplin Informatik, anderer Disziplinen der Informationswissenschaften, der technischen Anwendungen,

der wirtschaftlichen und gesellschaftlichen Auswirkungen u. a. vorgeschlagen und behandelt. Darin spiegeln sich vor allem die Neuheit des Fachs, die rasante Entwicklung der Technologien, die unterschiedlichen Einschätzungen von deren Bedeutung, der z. T. fehlende oder zuwenig konsistente bildungstheoretische Begründungsrahmen, aber auch ideologische Differenzen wider. Dies führte zu einer Palette möglicher Bildungsmassnahmen, von der sich angesichts beschränkter Stundenzahlen alle Bildungsinteressierten und -verantwortlichen jenes Stück nehmen konnten und können, das ihrem Verständnis und ihren Wünschen entspricht [...].

Zum methodischen Regelwerk, das auf der Grundlage fachimmanenter Eigenheiten und allgemeindidaktischer Erkenntnisse erstellt und in schulpraktischen Empfehlungen konkretisiert worden ist, kann festgehalten werden, dass die unterrichtliche Verwirklichung der informationstechnologischen Bildung starke Eigenheiten aufweist, die sie – entgegen anderen Aussagen – zu einem keinesfalls zu unterschätzenden sondern zu einem anspruchsvollen Unterfangen machen, wofür allgemeindidaktische Überlegungen und Erfahrungen aus anderen Fächern allein nicht mehr genügen. [...]“ [Eberle 1996, S. 429f]

### Forschungsfragen

Nach der Darstellung ausgewählter konstruktiver Elemente der Umsetzung informatischer Inhalte stellt EBERLE „Eine Auswahl offener Forschungsfragen“ vor. „Viele Aussagen in dieser Arbeit mit deskriptivem Charakter sind empirisch gar nicht oder mangelhaft nachgewiesen, beruhen nur auf Alltagsbeobachtungen oder basieren auf nichtrepräsentativer qualitativer Forschungsmethodik. Daraus ergibt sich eine breite Palette von Forschungsfragen, die empirisch geklärt werden sollten.“ [Eberle 1996, S. 427]

EBERLE hat mit seiner Habilitationsschrift einen Steinbruch an Quellen angehäuft, die in seiner Auswertung zu den oben angegebenen offenen Forschungsfragen führen. Der Bearbeitung dieser offenen Fragen wird die Didaktik der Informatik einen Teil ihrer Forschungen zu widmen haben. Durch unsere Studien wollen wir einen kleinen Beitrag dazu leisten. Auf dem Hintergrund der von EBERLE zu Recht eingeforderten empirischen Forschungsarbeiten ist es an der Zeit, neben qualitative Studien auch quantitative zu stellen, die die Chance eröffnen, aus vorhandenen Erfahrungen zu validen Aussagen in diesem Feld zu gelangen.

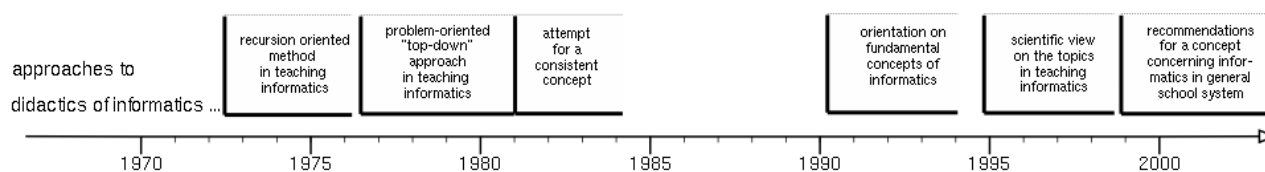


Abbildung 3: Zeitleiste zu didaktischen Orientierungen

## 4 Das Modulkonzept

### 4.1 Wissenschaftliche Fragestellungen

Folgende wissenschaftliche Fragestellungen sind Grundlage der Forschungsarbeit:

1. Zugänge zu größeren Problemklassen

Welche Zugänge erweisen sich als erfolgreich und zukunftsweisend, so dass die Chance besteht, dass Schülerinnen im Rahmen des Informatikunterrichts eine von aktuellen Moden unabhängige informatische Bildung erfahren?

2. Strukturierung der Fachinhalte unter fachdidaktischen Gesichtspunkten

Auf welche Weise kann eine Abfolge von Fachinhalten so strukturiert werden, dass nachhaltig informatische Bildung vermittelt wird?

3. Einfluss des Informatikunterrichts auf das Bild der Informatik bei Schülerinnen

Wie verändert ein bewusst auf die Zielsetzung „Generierung eines validen Bildes der Informatik als Wissenschaft“ orientierter Informatikunterricht die Sichtweise der Schülerinnen auf die Informatik? Welches Bild der Informatik wird im Informatikunterricht vermittelt?

Auf dem Hintergrund dieser forschungsleitenden Fragestellungen kann das folgende Forschungsprogramm abgeleitet werden:

Die Entwicklung und Evaluation von Konzepten ist konstitutives Element für die weitere Entwicklung der Fachdidaktik. Sie sollte sich auf empirische Untersuchungen stützen (vgl. Kap 3.2, S. 21). Vor der breiten Einführung von allgemein bildenden, verpflichtenden Informatikkursen sollte die Qualität des Unterrichts stärker als bisher betrachtet werden. Die vorgestellten Ansätze sollen eine nachhaltige Wirkung entfalten, damit die zu schaffende Basis nicht nur für aktuelle Lehr-/Lernprozesse umsetzbar ist, sondern den Schülerinnen Möglichkeiten eröffnet, grundlegende konzeptionelle Elemente der Informatik und ihrer Anwendung bei der Gestaltung der eigenen Zukunft berücksichtigen zu können.

Als zentrale Klammer eines problemlösenden Informatikunterrichts kommt der Informatischen Modellierung eine anerkannte Rolle in der Fachdidaktik Informatik zu. Dies zeigt nicht zuletzt die Ende 2000 von der Gesellschaft für Informatik in den Empfehlungen für ein Gesamtkonzept formulierte gleichlautende Leitlinie [Gesellschaft für Informatik 2000, S. 379]<sup>11</sup>. Bezüglich informatischer Modellierungsstrategien und -konzepte für den unterrichtlichen Umsetzungsprozess sollte die Zieldimension „Problemstellungen hinsichtlich verschiedener Lösungsstrategien resp. Modellierungen analysieren können“ für alle Schülerinnen der gymnasialen Sekundarstufe II Berücksichtigung finden.

Die propädeutische Ausrichtung von Kursen in der gymnasialen Oberstufe darf nicht außer Acht lassen, dass Schülerinnen vom Schulfach Informatik auch Hilfen für die Entwicklung eigener Perspektiven im Hinblick auf die Nutzung von Informatiksystemen erwarten. Dies führt dazu, dass die Klasse der Problemstellungen,

<sup>11</sup>Die weiteren Leitlinien sind

- Interaktion mit Informatiksystemen,
- Wirkprinzipien von Informatiksystemen,
- Wechselwirkungen zwischen Informatiksystemen, Individuum und Gesellschaft.

die im Unterricht konkrete Bearbeitung erfahren, nicht nur fachlich begründet, sondern auch an den Interessen der Schülerinnen orientiert werden muss. Bei einem anspruchsvollen, auf Problemlösen zielenden Informatikunterricht stellt die informatische Modellierung die Mittel zur Verfügung, mit deren Hilfe Problemstellungen bearbeitet werden.

## 4.2 Zur konzeptionellen Gestaltung – Vorstellung der Module

Zunächst gilt es, Entscheidungen zu treffen bezüglich des fachlichen Bereichs, der zu Beginn im Anfangsunterricht in der Sekundarstufe II bearbeitet werden soll. Um bei den Unterschieden zwischen Anspruch und Schulwirklichkeit produktive Lösungswege aufzuzeigen, ist es geboten, [neue] Konzepte umzusetzen und in ihrer Umsetzung einer Evaluation zu unterziehen. Als grundlegende Elemente eines modernen Informatikunterrichts wurden die informatische Modellierung, Rechnernetze und verteilte Informatiksysteme identifiziert. Die Umsetzung der Konzepte in konkreten Informatikunterricht erfolgt problemorientiert, d. h. ausgehend von einem Problembereich werden Probleme identifiziert, die unterrichtlich bearbeitet, gegebenenfalls modelliert werden. Dabei wird die fachdidaktische Orientierung an den von der Gesellschaft für Informatik vorgeschlagenen Leitlinien zu Grunde gelegt:

„Interaktion mit Informatiksystemen, Wirkprinzipien von Informatiksystemen, Informatische Modellierung, Wechselwirkungen zwischen Informatiksystemen, Individuum und Gesellschaft.

Die unter diesen Leitlinien strukturierten Kenntnisse und Fertigkeiten werden auf unterschiedlichem Niveau in der Primarstufe, in der Sekundarstufe I und in der Sekundarstufe II erworben, wobei stets an die Lebenswelt der Lernenden anzuknüpfen ist.“ [Gesellschaft für Informatik 2000, S. 379]

Die Strukturierung der in der Vermittlung bedeutsamen inhaltlichen Dimensionen sollte in enger Verzahnung mit informatischen Fachkonzepten vorgenommen werden. Allerdings darf in schulisch bildenden Prozessen nicht eine fachlich erschöpfende Bearbeitung von Informatikfachthemen erwartet werden. Im Gegenteil ist es anzuraten, die Bearbeitung fachlicher Themen auf verschiedenen Abstraktionsniveaus zu unterschiedlichen Zeitpunkten der schulischen Bildungsprozesse wieder aufzunehmen. Damit sind im Rahmen eines spiraligen Curriculums valide, didaktisch gestaltete Informatikinhalte zu vermitteln. Die quantitative Ausprägung der für die Umsetzung des Modulkonzepts verpflichtenden Anteile sollte auf Grund der aktiven und weiterhin sehr dynamischen Weiterentwicklung auf der Anwendungsseite auf einen beschränkten Teil der Unterrichtszeit im Schulfach Informatik beschränkt bleiben – dies sollte ca. ein Drittel der Unterrichtszeit sein.

Aktuell bedeutsamen Anwendungsbereichen, aber auch den persönlichen fachlichen Stärken und Interessen der Schülerinnen und der Unterrichtenden, muss im Schulfach Informatik eine Entfaltungsmöglichkeit gegeben werden. Dies gilt vor allem, sobald in projektorientierten und fachübergreifenden Zusammenhängen gearbeitet wird. Das zweite Drittel ist damit dem Anwendungszusammenhang auf informatischer Basis zuzugestehen. Die Arbeit in vernetzten Strukturen setzt soziale Fähigkeiten voraus, denen im Zusammenhang mit schulischen Aneignungsprozessen explizit Unterrichtszeit zugestanden werden muss.

Um die fachlichen Anforderungen zu strukturieren, bietet sich ein modularisiertes Konzept an. Bei der Sequenzierung ist zu bedenken, dass die Arbeit der Schülerinnen mit dem konkreten schulischen Intranet auf einer informatischen Fachbasis zu erfolgen hat. Daraus folgt, dass dem Modul „Informatiksysteme verstehen und verantwortlich nutzen“ eine prioritäre Rolle zugestanden werden muss. Elemente dieses Moduls müssen vor Beginn der Arbeit mit den schulischen Informatiksystemen erschlossen werden.

Der Berücksichtigung der historischen Dimension der Module sollte integriert Rechnung getragen werden. Elemente der Geschichte der Informatik, vergleichende Überlegungen im Zusammenhang mit Ausprägungen für konkrete Implementierungen, programmiersprachliche Konstrukte, sowie Überlegungen zu graphischen Benutzungsoberflächen und zum Recht auf informationelle Selbstbestimmung sollten im Zusammenhang mit konkreten Problemstellungen und Modellierungen thematisiert werden. Ausgangspunkt des Informatikunterrichts sind Problemstellungen, die formuliert werden und zu denen nach erfolgter Modellierung reflektierend zurückgekehrt wird. Damit stellt die Problemorientierung die methodische Klammer des Informatikunterrichts dar.

#### **4.2.1 Informatiksysteme verstehen und verantwortlich nutzen**

Die zunehmende Nutzung komplexer vernetzter Informatiksysteme im schulischen Zusammenhang sollte dazu führen, dass die Hintergründe für Regelungen in den Blick genommen werden. Die erweiterte Nutzung hat zur Folge, dass unabdingbare Regelungen für Räume, Informatiksysteme und vernetzte Systeme aufgestellt werden (durch die Systembetreuung, die Schulgemeinde, die Schulträger, den Gesetzgeber). Diese Regelungen basieren auf dem Schutz der Persönlichkeit und der Möglichkeit, ein Leben in eigener Verantwortung zu gestalten und der Vermeidung unnötiger Belastung und Schädigung der technisch-administrativen Infrastruktur. Bei allen Nutzerinnen muss soviel Hintergrundwissen vorhanden sein, dass die Umsetzung dieser Anforderungen nicht durch „Gehorchen und Befolgen“, sondern durch Einsicht in die gesellschaftlichen und technischen Hintergründe (Themenbereich Rechnernetze und Verteilte Systeme) ermöglicht wird. Bei der Arbeit mit elaborierten Lernumgebungen (CSCL-Systeme), die zunehmend Bestandteil schulischer Intranetstrukturen sind, müssen Rechte Dritter beachtet werden. Durch Einsicht in schutzwürdige Belange (Netiquette) kann z. B. das Veröffentlichen oder Kopieren von geschützten Materialien wirksamer verhindert werden als ausschließlich durch Verbote.

Wenn Betriebssysteme handlungsorientiert erschlossen werden, ist dafür Sorge zu tragen, dass jedes Element auf der Folie der Verallgemeinerung und Zukunftssicherheit eine unterrichtliche Umsetzung erfährt. Dies kann z. B. dadurch erreicht werden, dass Betriebssystemfunktionen über betriebssystem- und plattformunabhängige [Skript-] Sprachen für explorative Unterrichtseinheiten genutzt werden. Dies eröffnet damit gleichzeitig die Erarbeitung curricularer Elemente, die hochgradig portabel sind und es den Schülerinnen erlauben, erarbeitete Elemente zur Automatisierung von Betriebssystemfunktionen zu nutzen.

Mit der Entwicklung graphischer Benutzungsoberflächen ist ein aktueller Stand zur Benutzung von Informatiksystemen erreicht, der in den nächsten Jahren eine Erweiterung erfahren wird. Soll Informatikunterricht zukunftsweisende Impulse setzen, so ist hier durch die Trennung der verschiedenen Ebenen bei der Modellierung von Informatiksystemen z. B. mit dem MVC-Konzept (aber auch anderen Entwurfsmustern) eine Möglichkeit aufzuzeigen, wie zukünftige Entwicklungen berücksichtigt werden können, ohne erarbeitete und verstandene Fachkonzepte obsolet werden zu lassen.

#### **4.2.2 Modellierung - zentrales Tätigkeitsfeld informatischer Arbeit**

Bei der Modellierung gilt es herauszuarbeiten, wozu die Modellierung vorgenommen wird. Die Modellierung ist zur Exploration und Deskription des existierenden Informatiksystems der Schule am konkreten Schülerarbeitsplatz hilfreich. Damit kann die Verbindung zu dem Modul *Informatiksysteme verantwortlich nutzen*



und verstehen hergestellt werden. Mit der Modellierung sollte die Möglichkeit eröffnet werden, über motivierende Probleme hinaus Klassen von Problemen zu bearbeiten. Es ist notwendig, Unterschiede zwischen der Modellierung in der Informatik und der Modellierung in anderen technisch-wissenschaftlichen Gegenstandsbereichen durch Rückwirkung des Modells „auf die Welt“ deutlich werden zu lassen. Als Ergebnis (Produkt) der informatischen Modellierung werden z. B. Arbeitsabläufe nicht nur theoretisch, sondern konkret beeinflusst, verändert oder gar zu grossen Teilen automatisiert.

Es gibt nicht nur eine informatische Modellierung, sondern verschiedene, die sich dadurch auszeichnen, dass der Gegenstandsbereich jeweils durch eine besondere „Brille“ betrachtet wird, um besonders effektive Modellbildungsstrategien anwenden zu können. Die dadurch vorgenommene Reduktion sollte im konkreten Fall thematisiert werden. Schülerinnen kann die durch die jeweilige Modellierung „abgeschnittene“ oder „ausgeblendete“ Sicht verdeutlicht werden, indem z. B. eine konkrete Problemstellung mit verschiedenen Sichten betrachtet und bearbeitet wird. Methoden und Veranschaulichungen, die auf eine ganz spezielle Modellierung beschränkt sind, haben nur einen begrenzten Zukunftswert. Heute werden umfangreiche Probleme nicht mehr mit SADT oder der Jackson-Methode „behandelt“. Andererseits haben ERM oder Und-Oder-Bäume weiterhin ihren Platz bei bestimmten Anwendungsfällen. Hier ist von Fall zu Fall zu entscheiden, welche konkrete Darstellung als sinnvoller und notwendiger Beitrag zur Modellierung unterrichtlich verankert werden sollte. Allerdings sollte eine Orientierung an fachlichen Standards vorgenommen werden, wobei eine sinnvolle didaktische Gestaltung angestrebt werden muss.

### **Erkenntnisse der theoretischen Informatik im Anwendungskontext**

Theoretische Überlegungen der Informatik haben unmittelbare Konsequenzen. Sie sind notwendig zum Verständnis grundlegender Wirkprinzipien von Informatiksystemen, lassen sich einsetzen, um Strukturen zu verdeutlichen und zu modellieren. Die Verbindung mit nicht-naturwissenschaftlichen Fachinhalten kann über das gemeinsame Feld der Kategorisierung von Sprachen erschlossen werden. Fragen, die die Zukunft der Schülerinnen objektiv betreffen, berühren die Komplexität möglicher Informatiksysteme, die Chancen der Sprachübersetzung natürlicher Sprachen, die prinzipiellen Grenzen informatischer Modellierung und bieten sich für fachübergreifende Ansätze aus dem Bereich der Philosophie geradezu an.

### **Wissensbasierte Modellierung**

Nach einer Phase der Unklarheit kann festgestellt werden, dass unter den Informatikerinnen inzwischen Einigkeit über die Definition einiger zentraler Begriffe hergestellt wurde, die sich nicht exakt definieren lassen und sicher auch weiterhin kontrovers diskutiert werden: Daten, Wissen und Information. Für Schülerinnen sollte deutlich werden, dass es notwendig ist, einen validen Begriffsapparat zu benutzen, der es gestattet, in einer Gemeinschaft einen Inhaltsbereich gestaltend zu erschließen. Die zum Standardrepertoire der Informatik gehörenden Begriffe Syntax, Semantik und Pragmatik stellen einen informatischen Zugang zur Bestimmung dieser Begriffe bereit.

Im Zusammenhang mit der zunehmenden Bedeutung großer Datensammlungen (z. B. im Zusammenhang mit dem Internet) ist die Akquisition von Wissen leichter denn je. Die Modellierung muss von der Anwendung auf die dahinter liegenden Strukturen und Modellvorstellungen verlagert werden. Damit kann z. B. aus der Black-Box Suchmaschine eine White-Box entwickelt werden. Allerdings sind gerade die bei der konkreten Modellierung auftretenden Schwierigkeiten nicht ohne beachtlichen unterrichtlichen Aufwand zu lösen. Der besondere Variablenbegriff der wissensbasierten Modellierung bedarf der ausführlichen Fundierung, sollen

die Erkenntnisse nicht an der Oberfläche bleiben. In diesem Zusammenhang sollten die gesellschaftlichen Rahmenbedingungen für die Arbeit mit großen Datenbeständen beispielhaft illustriert werden: Persönlichkeitsschutz, Datenschutz und Datensicherheit – zur interessengeleiteten Bewertung des Rechts auf informationelle Selbstbestimmung und ihrer rechtlichen Dimension. Der Gewinnung von Daten und dem Gewinnen von Information sind die beiden Endpunkte der Nutzung von Informatiksystemen zuzuordnen – zum Einen in Richtung auf die Vorbereitung der maschinellen Verarbeitung und zum Anderen in Richtung auf die Benutzung eines Informatiksystems.

### **Objektorientierte Modellierung**

Die objektorientierte Beschreibung von konkreten Informatiksystemen liefert ein Modell für die erfolgreiche Implementierung von Funktionalität. Inzwischen existiert ein zunehmender Fundus an Ideen für Problemstellungen, die sich im Schulfach Informatik für die Objektorientierte Modellierung anbieten. Im Zusammenhang mit dem Teilgebiet Rechnernetze und Verteilte Systeme, mit ereignisgetriebenen Systemen, mit graphischen Benutzungsoberflächen, mit Interaktion und Kommunikation und nicht zuletzt mit Simulation kommt der Objektorientierten Modellierung eine hervorragende Bedeutung zu, die unterrichtlich erfolgreich umgesetzt werden kann. Allerdings sind die verpflichtenden Schwerpunkte: einfache Algorithmen und Datenstrukturen und der Variablenbegriff explizit zu thematisieren. Oftmals wird bei der Konstruktion neuer curricularer Elemente zu wenig berücksichtigt, dass notwendige Voraussetzungen aus dem Fundus der bekannten Modellierungen weiterhin berücksichtigt werden müssen und vor allem ihren zeitlichen Tribut im Unterrichtsprozess fordern.

Aus dem Bereich der Objektorientierten Modellierung bieten sich an mehreren Stellen Übergangsmöglichkeiten zu anderen Modellierungen an: z. B. kann mit der Anbindung von Datenbankschnittstellen auf umfangreiche Datenbestände zugegriffen werden; außerdem sind Erweiterungen verfügbar, die die Nutzung von Elementen der funktionalen Modellierung erlauben. Als fakultative Schwerpunkte im Zusammenhang mit der Objektorientierten Modellierung bieten sich die Bereiche Nebenläufigkeit, Dokumentenbeschreibungssprachen (Äquivalenz von Dokumenten- und Datenstruktur) geradezu an.

### **Funktionale Modellierung**

Funktionen als Argumente von Funktionen sind neben speziellen Möglichkeiten der Anwendung von Operationen auf Listen von Funktionen die zentralen Elemente der funktionalen Modellierung. Anwendungsfälle für den Informatikunterricht krankten bisher an der Nähe zur Mathematik, die von vielen Kolleginnen und Schülerinnen nicht positiv annotiert wurden. Es sollte der Versuch unternommen werden, die Integration funktional modellierter Elemente an den Stellen zu unternehmen, die im Modellierungsprozess sinnvoll und angemessen sind.

### **Anmerkungen zum Modulkonzept**

Die vorgestellten Module stellen einen Rahmen dar, der nicht mit einer Sequenz verwechselt werden darf. Bei der Gestaltung konkreter Kurse ergeben sich Verzahnungsmöglichkeiten zwischen verschiedenen Modulen. Beispiele für modulübergreifende Fragestellungen:

- Mit der prädikativen Modellierung ist es möglich, das Konfigurationsmanagement von Betriebssystemen unterrichtlich zu untersetzen.

- Die Attribute von Dateien können nicht nur über die vom konkreten Betriebssystem zur Verfügung gestellten Anzeigemöglichkeiten, sondern auch über plattformübergreifende und -unabhängige [Skript-] Sprachen herausgefunden werden.
- Mit der Thematisierung von Internet-Protokollen können automaten-theoretische Elemente in den Unterrichtszusammenhang eingebunden werden.

Eine Verschränkung der Module in der Gestaltung des Unterrichts ist notwendig. Die Ausprägung der inhaltlichen Umsetzung variiert mit der konkreten Problemstellung, mit der sich die Schülerinnen modellierend, handelnd, gestaltend und reflektierend auseinandersetzen.

### 4.3 Konkretisierung – Dokumentation von Kursbeispielen

Da keine fachlich fundierten Kenntnisse aus der Sekundarstufe I bei den Schülerinnen zu Beginn des 11. Jahrgangs vorausgesetzt werden können, besteht die Notwendigkeit, die unabdingbaren, grundlegenden Kenntnisse integriert in den Kursverlauf einzubinden, um darauf Bezug nehmend die Herausbildung informatischer Kompetenzen zu erreichen. Die unterrichtliche Beschäftigung mit dem Thema CSCL (Computer Supported Cooperative Learning) sollte integriert erfolgen. 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 Informatikinhalt eingebunden. Ein Paradigmenwechsel in den Grundkursen Informatik sollte unterrichtlich bereits im 11. Jahrgang umgesetzt werden, damit die Schülerinnen mindestens ein weiteres Modellierungskonzept (z. B. prädikative Modellierung) der Informatik kennen lernen.

Die methodische Orientierung des Informatikunterrichts sollte den beiden Dimensionen: Problemorientierung und Projektorientierung entsprechen. Konkret bedeutet dies: Schülerinnen erarbeiten an Hand von Problemstellungen, die im Laufe des Unterrichts zunehmend komplexer und offener zu gestalten und zu formulieren sind, die je notwendigen Fachinhalte. Im Zentrum der unterrichtlichen Arbeit stehen damit konkrete Probleme, die aber nicht so gestaltet werden, dass z. B. an Hand der Problemstellung etwa Sprachelemente „erledigt“ werden.

Die Orientierung an fachlichen Konzepten ermöglicht in der Phase der Problemgenerierung und -spezifikation eine von der konkreten Umsetzung in erster Näherung unabhängige Arbeit. Dies setzt bei den Unterrichtenden Überblickswissen voraus, damit mögliche Lösungswege der Schülerinnen eingeschätzt werden können. Allerdings ist die Entscheidung über die Modellierungsmethode zu Beginn von den Unterrichtenden so zu treffen, dass den Schülerinnen eine Orientierung ermöglicht wird. Erst nachdem die Schülerinnen über grundlegende Kenntnisse bezüglich der Modellierung verfügen, kann eine weitere Modellierungsmethode sinnvoll thematisiert werden.

Auf dem Hintergrund dieser Vorüberlegungen sind unterrichtliche Sequenzen gestaltet worden, die außerdem den schulischen Rahmenbedingungen (Richtlinien und Kooperation mit den Fachkollegen, da die parallelen Informatikkurse nach dem 11. Jahrgang zusammenschmelzen und nur ein Kurs weitergeführt wird) genügen.

Die geplante Unterrichtssequenz musste weiterhin mit der technisch im Aufbau befindlichen schulischen Intranetstruktur zur Passung gebracht werden:

- Beispiel: Die für den CSCL-Strang notwendige Serversoftware konnte erst während der ersten Erprobung im schulischen Intranet installiert werden, so dass der gewünschte integrative Einsatz nicht von Beginn an, sondern erst nach einiger Zeit realisiert werden konnte.

Der konkrete Gestaltungsvorschlag für ein schulinternes Curriculum (Sequenzierung: Grundkurs Informatik ab Schuljahr 1999/2000) ist im Anhang A, S. 33 beigelegt und befindet sich zur Zeit in Erprobung.

#### 4.4 Untersetzung der Fragestellungen in Untersuchungsvorhaben

Zu Untersuchungen im Zusammenhang mit der *Fragestellung 1* (siehe 4.1, Seite 23):

Modellierung als fachdidaktischer Rahmen des Informatikunterrichts eröffnet eine breite Palette von Möglichkeiten. Als Beispiele seien genannt: Wissensbasierte Modellierung, Modellierung von Automaten, Objektorientierte Modellierung, Rechnernetze und verteilte Systeme. Mit dem Ansatz Modellierung von Automaten wurden bereits umfangreiche Erfahrungen gesammelt und dokumentiert (vgl. z. B. [Humbert 1999]).

##### Hilfen für den Unterricht

Im Zusammenhang mit konkreten Problemstellungen sollen Modellierungsstrategien und Fachkonzepte erarbeitet und eingesetzt werden. Die selbstständige Arbeit der Schülerinnen zur Modellierung und Lösung einer konkreten Problemstellung benötigt eine Grundlage, die an verschiedenen Beispielen erlernt und geübt werden muss. Zu jeder Modellierung gehören sowohl sprachliche, wie auch graphische Vereinbarungen, die es erlauben, den Modellierungsprozess in geeigneter Weise darzustellen und abzubilden, und damit verschiedenen Lerntypen gerecht werden. [Jochum 1998] verdeutlicht den Stellenwert graphischer Hilfsmittel bei der Modellierung im Informatikunterricht und zeigt den Einsatz von CRC-Karten zur Unterstützung kommunikativer Prozesse auf der Ebene der Lerngruppe. Im Zusammenhang mit Objektorientierter Modellierung bieten sich ausgewählte Elemente der Beschreibungssprache UML als eine graphische und standardisierte Form der Darstellung an, die den gesamten Modellbildungsprozess begleiten.

Für die Untersuchungen zur Konkretion und Sequenzierung (*Fragestellung 2* siehe 4.1, Seite 23) wurde der Anfangsunterricht in Informatikgrundkursen gestaltet und prozessbegleitend evaluiert. Dazu wurden u. a. ausgewählte Unterrichtsstunden des Informatikunterrichts der Jahrgangsstufe 11 zur objektorientierten Modellierung mit Hilfe der Videoanalyse dokumentiert und untersucht. Die Analyse zeigt, dass ausgewählte Elemente der graphischen Beschreibung zur Unterstützung der Orientierung im Lehr-/Lernprozess nützlich sind. Die Abstraktionsleistungen werden im dokumentierten Unterrichtsprozess von den Schülerinnen erbracht. Es wird deutlich, dass der Modellierungsprozess neben der Abstraktionsleistung eine begriffliche Sicherheit erfordert. Diese begriffliche Exaktheit konnte von den Schülerinnen nur mit Mühe erreicht werden: bei Lernerfolgskontrollen zeigt sich, dass Unsicherheiten in der Definition von zentralen Begriffen, wie Klasse und Objekt, Attribut und Wert eines Attributs, allgemeines Fachkonzept und konkrete Umsetzung besteht.

##### Bild der Informatik bei Schülerinnen

*Fragestellungen 1 und 3* (siehe 4.1, Seite 23):

Empirischer Rahmen für die durchgeführten Untersuchungen zum Bild der Informatik bei Schülerinnen: Gemessen wird die „Einschätzung der Informatik durch Lernende“. Als Kontrollgröße ist die fachdidaktische

Orientierung „Informatisches Modellieren mit historischer Reflexion“ als Grundlage des Unterrichts anzusehen. Der intervenierende Faktor ist in den aktuellen Regeln der Abiturzulassungsordnung zu sehen. Zur Datenerhebung wurden Befragungen mit Fragebögen von Schülerinnen des 11. Jahrgangs in fünf Informatikgrundkursen im Laufe des Schuljahres 1999/2000 durchgeführt und im Rahmen eines Längsschnitts im Schuljahr 2000/2001 fortgesetzt. Nach einer ersten Auswertung von Fragebögen eines Informatikkurses wurden in einem Gruppeninterview (Leitfadeninterview) im Februar 2000 ausgewählte Schülerinnen zum Informatikunterricht befragt. Die Ergebnisse dieser Untersuchungen (Fragebogenauswertung und Analyse des Gruppeninterviews) lassen sich wie folgt zusammenfassen:

Grundlage war die Frage nach der Herausbildung eines Bildes der Informatik bei den Schülerinnen. Das Bild entwickelt sich sowohl durch den Informatikunterricht als auch auf dem Erfahrungshintergrund der Schülerinnen außerhalb des Informatikunterrichts. Dem Informatikunterricht kommt eine wichtige Bedeutung zu, indem er Korrektiv für falsche Vorstellungen, Erwartungen und Ängste ist. Es konnte herausgefunden werden, dass Schülerinnen z. B. Sorge vor der Zukunft zum Ausdruck bringen, weil sie sich nicht in der Lage sehen, abzuschätzen, dass es prinzipiell unlösbare Problemklassen gibt. Im Hintergrund einiger Antworten stehen Einschätzungen über die individuelle und gesellschaftliche Zukunftsbedeutung der Informatik, die durch verhaltene Hoffnungen, aber auch durch Unsicherheit geprägt sind.<sup>12</sup>

Zusammen mit einem validen Bild der Wissenschaft Informatik werden für die Schülerinnen zukünftige Entwicklungen einschätzbar. Damit erfährt die Herausbildung des Bildes eine Zukunftsbedeutung für die individuelle Lebensgestaltung. Für die hier (zusammenfassend) wiedergegebenen Ergebnisse wurden fünf Informatikgrundkurse befragt (Fragebögen) - mit einem halben Grundkurs wurde ein leitfadengesteuertes Gruppeninterview durchgeführt und transkribiert. Im Längsschnitt werden einige Schülerinnen weiter untersucht. Dies soll Aussagen über die Änderung des entwickelten Bildes der Informatik als Wissenschaft bei den Schülerinnen durch den Informatikunterricht ermöglichen und kann so eine Planungsbasis für weitere Untersuchungen darstellen.

### Expertinneninterviews

Zur weiteren Untersuchung zur Exploration bezogen auf die *Fragestellungen 1 und 2* (siehe 4.1, Seite 23): wurden Interviews mit Informatiklehrerinnen<sup>13</sup> durchgeführt. Die Auswertung dieser Befragungen ist noch nicht abgeschlossen.

Im Folgenden werden erste Ergebnisse skizziert:

Bei der handlungsorientierten Umsetzung von Fachkonzepten in konkrete, ablauffähige Ergebnisse treten bei vielen Schülerinnen Probleme auf. Diese liegen nach Einschätzung der Befragten u. a. darin begründet, dass eine gewisse Routine im Umgang mit konkreten Informatiksystemen eine notwendige Voraussetzung für die erfolgreiche Arbeit im Informatikunterricht darstellen. Das beginnt bei der Bedienung der konkreten in der Schule zur Verfügung stehenden Informatiksysteme, ihrer Benutzungsoberfläche und ihren Nutzungsbedingungen. Es setzt sich über die arbeitsökonomische Bedienung eines Texteditors oder eines Entwicklungssystems bis hin zur Kenntnis der Syntax einer bestimmten Programmiersprache fort. Dazu zählen auch Kenntnisse bezogen auf Hilfsmöglichkeiten für die Fehlersuche im eingegebenen Quellcode, seien es nun syntaktische

<sup>12</sup>Hierbei sollte berücksichtigt werden, dass in dieser Altersstufe bei Jugendlichen/Heranwachsenden unabhängig von konkreten Unterrichtsfächern eine Orientierung auf ihre je individuelle sich entwickelnde „Weltsicht“ festgestellt werden kann (vgl. [Fischer u. a. 2000]).

<sup>13</sup>Insgesamt wurden 16 Interviews durchgeführt (20 Informatiklehrkräfte sollten befragt werden)

oder durch unangemessene Modellierung verursachte Fehler. Diese zum Teil komplexen Voraussetzungen können nicht sämtlich nebenbei oder integriert erworben werden. Der Weg aus dieser unbefriedigenden Situation besteht nach Meinung einiger der interviewten Kolleginnen darin, mit Informatikwerkzeugen zu arbeiten, die die Modellierung unterstützen und Rahmen für die Umsetzung der Modellierung in ablauffähige Programme liefern, die dann gefüllt werden müssen.

Der fachliche Paradigmenwechsel zur Objektorientierten Modellierung befindet sich in der Phase der Konsolidierung und beginnt, seine umfassende Struktur nach und nach zu entfalten. Die interviewten Lehrerinnen setzen zunehmend auf „Offene Standards“. Einige Rückmeldungen lassen deutlich werden, dass „moderne Programmiersprachen“ auf ihren didaktischen Wert untersucht werden sollten, bevor sie für den schulischen Einsatz propagiert werden.

## 5 Ergebnisse – Perspektiven

Im Forschungszusammenhang wurden und werden Unterrichtsreihen praktisch umgesetzt und prozessbegleitend evaluiert. Neben Verfahren zur Lernerfolgskontrolle und Leistungsbewertung werden weitere Möglichkeiten zur Evaluation auf dem Hintergrund der Forschungsfragen eingesetzt. In der Unterrichtsforschung ist nicht alles nur durch eine Erhebungstechnik adäquat erfassbar. Jede Methode hat ihre spezifischen Stärken und Schwächen, so dass in der Literatur zur Evaluationsforschung ein Methodenmix empfohlen wird.

Bei der Umsetzung konzeptioneller Elemente zur Gestaltung des Informatikunterrichts wurden die Überlegungen durch Diskussionen und unterrichtspraktische Erfahrungen im Zusammenhang mit der Ausbildung der Informatikstudienreferendarinnen im Studienseminar für das Lehramt für die Sekundarstufe II in Hamm erheblich bereichert. Ergebnisse der Ausbildung wurden als Beitrag zur Entwicklung der Informatiklehrerinnenausbildung dokumentiert (siehe [Humbert u. a. 2000] und [Rickert u. a. 2001]).

Aus den Ergebnissen der bisher durchgeführten Studien läßt sich ein großer Bedarf nach weiteren Studien ableiten, der nicht im Rahmen dieser Forschungsarbeiten bewältigt werden kann. Schwerpunkte sollten dabei in einer breit angelegten Studie über die „Wirksamkeit des Informatikunterrichts“ in der gymnasialen Oberstufe, aber auch in der Längsschnittuntersuchung von Schülerpopulationen gesetzt werden. Dabei sollte ein besonderes Augenmerk auf die Evaluation der Team- und Kooperationsfähigkeit der Schülerinnen gelegt werden.

In der Untersuchung zur Analyse und Evaluation der Softwareentwicklung in Deutschland für das Bundesministerium für Bildung und Forschung (BMBF) wird im Kapitel 9 „Schlussfolgerungen“ für den Bereich Aus- und Weiterbildung festgestellt:

„Zur Steigerung der Attraktivität des Standorts Deutschland für software-intensive Branchen muss kurz-, mittel- und langfristig die Verfügbarkeit einer ausreichenden Anzahl von qualifizierten Arbeitskräften auf allen Qualifikationsebenen gesichert sein. [...] Mittel- und langfristig kann das Personalproblem nur durch eine strukturelle Änderung der traditionellen Ausbildung [...] selbst, sowie deren Integration mit anderen Aus- und Weiterbildungsangeboten im Sinne eines lebenslangen Lernmodells nachhaltig gelöst werden. Somit müssen konkrete Maßnahmen folgendes beinhalten: [...] Bereits frühzeitig muss die Bereitschaft zur Auseinandersetzung mit Themen der Informationstechnologie und ein selbstverständlicher Umgang mit diesen Kommunikationsmedien gefördert werden. Diese Grundlagen müssen bereits in den Schulen flächendeckend vermittelt werden [...]. Dieser Forderung ist auch durch entsprechende Maßnahmen bei der Aus- und Weiterbildung

von Lehrern Rechnung zu tragen. [...] Es gibt wohl in keinem Bereich ähnlich verzerrte Vorstellungen über die Berufsbilder wie in der Informatik. In der breiten Öffentlichkeit werden mit dem Berufsbild Informatiker oder Softwareentwickler immer noch technikbegeisterte Programmierer verstanden. Unter diesem Eindruck kann es nicht verwundern, dass insbesondere Frauen sich nur sehr begrenzt für ein derartiges Studium interessieren.“ [GfK Marktforschung GmbH u. a. 2000, S. 185ff - Zitatencollage]

## Anhang

### A Gestaltung eines Grundkurses Informatik (11., 12. Jhg.)

#### 11. Jhg.:

- Vorbereitung:  
Sammlung von Vorstellungen der Schülerinnen über die Wissenschaft Informatik  
Vorstrukturierung der genannten Begriffe
- Rechnernetze und verteilte Systeme  
Einführung in die produktive Arbeit mit dem schulischen Intranet (dies muss m. E. in Zukunft und didaktisch gestaltet in den verpflichtenden Teil des Informatikunterrichts der Sekundarstufe I integriert werden)  
Da Unterrichtszeit des 11. Jahrgangs auf die Erarbeitung der Voraussetzungen für die Arbeit mit den schulischen Informatiksystemen verwendet werden muss, ist die Thematisierung des Themenbereichs Rechnernetze und verteilte Systeme mit dem Ziel des Erwerbs von Handlungskompetenzen zur effektiven Nutzung der schulischen Intranetinfrastruktur unabdingbar.
  1. personenbezogenes Accounting im schulischen Intranet und seine Konsequenz: Vertrauen ist gut, Kontrolle ist besser?  
Überwachungsmöglichkeiten
  2. Erläuterungen zur Nutzung der serverbasierten Profilaufbewahrung im schulischen Intranet
  3. praktische Übungen:
    - (a) Einlogvorgang,
    - (b) Schreibtischmetapher für die Nutzung von Benutzungsoberflächen von Informatiksystemen
      - i. Organisationsmöglichkeiten – Anlegen von Ordnerstrukturen
      - ii. Hinweise auf explorative Arbeitsweisen (Nutzung der Hilfefunktion, Kurzbefehle, besondere Zeichen/Tasten)
    - (c) Starten und Beenden von Programmen
  4. Netiquette und ihre informatischen Hintergründe  
Was hat Ethik mit Technik zu tun?
  5. Einführung in die Arbeit mit dem Mailprogramm  
allgemeine Funktionen
    - (a) Empfang von Mails
    - (b) Senden von Mails
    - (c) Organisationsmöglichkeiten für Ordnerstrukturen am Beispiel von Mailordnern
      - Hierarchie: flache vs. tiefe Schachtelung – Pro und Kontra
- elektronisches Publizieren (für Übungen wird ein einfacher Texteditor eingesetzt – die Ergebnisse werden in einem Standardbrowser betrachtet)



1. Dokumentenbeschreibungssprachen: Zweck und Ziel
  2. von SGML nach HTML: zur geschichtlichen Entwicklung
  3. Klammerstrukturen als Beispiel für syntaktische Strukturen
  4. Struktur von Texten und ihre Repräsentation in HTML
    - Tags: html, head, body, header ( $h_i$  mit  $i \in \{1 \dots 6\}$ ), paragraph
  5. Einbindung von Bildern in Texte
    - Tag: image (img) – Attribute: source (src), width, height, border
  6. Strukturieren von Dokumenten durch Verweise - Vernetzen
    - Tag: Anchor (a) – Attribut: href="<url>"<sup>14</sup>
  7. Zur Strukturierung von Dokumenten mit Hilfe von Verweisen
    - tiefe vs. flache Hierarchie
  8. HTML-Dokumente und Netiquette
    - Tag: anchor – Attribut: href="mailto:<mailadresse>"<sup>15</sup>
    - Überlegungen zu minimalen Anforderungen, die an HTML-Seiten zu stellen sind, führen zu einem von den Schülerinnen erarbeiteten Gestaltungsvorschlag für einen grundsätzlichen, minimalen Rahmen
  9. Strukturierung innerhalb von Dokumenten: Listen und Tabellen
  10. die weitere Entwicklung von Dokumentenbeschreibungssprachen:
 

Normen: Vereinbarungen/Veränderungen/Erweiterungen (Doctype)
- Computer Supported Cooperative Learning
    - erweiterte Schreibtischmetapher als Ausgangspunkt für Überlegungen zur kollaborativen Arbeit in vernetzten Strukturen
    - Unterrichtsplanung:
 

B.3 Einführung in die Arbeit mit einem webbasierten CSCW zur Kursplanung, S. 38
    - Strukturüberlegungen: Anpassung der Bereiche an die Nutzungsanforderungen, die durch die Analyse der Struktur der Aufbau- und Ablauforganisation eines konkreten Arbeitsprozesses gefunden werden
  - informatisches Modellieren am Beispiel der Objektorientierung
 

Grundlegende informatische Konzepte werden an Hand von nicht trivialen offenen Problemstellungen erarbeitet

integriert in die objektorientierte Modellierung wird die Erstellung von Algorithmen unterrichtlich umgesetzt

#### 1. Unterschiedliche Arten der Modellierung:

<sup>14</sup>an dieser Stelle wird `<url> := http://<rechnername>/<pfad>/<dokument>` betrachtet und benutzt

<sup>15</sup>an dieser Stelle erfolgt die Verallgemeinerung der Syntax für `<url>` gemäß **BERNERS-LEE, Tim**: RFC 1630 - Uniform Resource Identifiers in WWW. <http://www.faqs.org/rfcs/rfc1630.html> June 1994

- (a) Strukturierung von Texten als Modellierung,
- (b) Verbinden und Anordnen von Dokumenten in Strukturen als Modellierung,
- (c) allgemeiner Modellbegriff und Ausprägungen in der Informatik

## 2. Einführung in die objektorientierte Modellierung

- (a) das konkrete Informatiksystem als Grundlage für die Modellierung  
Unterrichtsplanung:  
B.1 Objektorientierte Modellierung am Beispiel des Schülercomputerarbeitsplatzes, S. 36
- (b) Modellierung graphisch orientierter Sachverhalte durch die Nutzung vorgegebener Klassen,
- (c) Ereignisse als Kennzeichen grafischer Benutzungsoberflächen und ihre Nutzung in der Modellierung interaktiver Informatiksysteme,
- (d) Modellierung einer offenen Problemsituation (Ampelanlage)  
Unterrichtsplanung:  
B.2 Modellierung grundlegender Elemente für ein Blinklicht, S. 37
- (e) Modellierung einer analogen Uhr – das Problem der Nebenläufigkeit

## 12. Jhg.:

- Fachsystematik
  - effiziente Organisation von Daten mit Hilfe von Datenstrukturen  
Unterrichtsplanung:  
B.4 Das Fachkonzept Keller - Realisierung als Klasse, S. 42
  - Server-/Klienten-Modellierung  
Unterrichtsplanung:  
B.5 Modellierung von Server-/Clientstrukturen, S. 45
- Anwendungsbereich
  - Auswahl und Modellierung konkreter Problemstellungen
  - Präsentationsunterstützung durch strukturierte Dokumentation
- Computer Supported Cooperative Learning
  - Awareness (als Funktion)
  - Berichtswesen
  - Erarbeitung der notwendigen Grundlagen, um ein elektronisches Portfolio anzulegen
  - Anlegen eines Portfolios

## B Ausgewählte Unterrichtsplanung und -durchführung

### B.1 Objektorientierte Modellierung am Beispiel des Schülercomputerarbeitsplatzes

#### Einordnung: Objektorientierte Modellierung – Klassenstruktur

##### Stundenlernziel

Die Schülerinnen erhalten einen Einblick in die objektorientierte Beschreibung des Computerarbeitsplatzes.

##### kognitive Zieldimension:

Die Schülerinnen

- beschreiben ihren Schülercomputerarbeitsplatz mit Hilfe einer einfachen Klassenstruktur, bestehend aus *Bildschirm, Tastatur, Maus*.
- entdecken, dass eine Klasse (*Stift*) nötig ist, um den Bildschirminhalt bearbeiten zu können.
- beschreiben vermutete Eigenschaften (Attribute) und Funktionen (Methoden), die die Klassen *Bildschirm, Tastatur, Maus* und *Stift* für die Arbeit haben können/sollen.
- füllen eine grafische Klassenbeschreibung mit den Attributen und Methoden.

##### affektive Zieldimension:

Die Schülerinnen

- entdecken, dass der Computerarbeitsplatz mit objektorientierten Begriffen beschrieben werden kann.

#### Geplante Verlaufsstruktur

Phase	Inhalte	Form/Medien
Einstieg	Präsentation: Abbildung eines Computerarbeitsplatzes	Abbildung/Beamer
Problem	Impuls: Modell bilden - Beschreiben - Klassifizieren	
Erarbeitung	Bildung von Funktionseinheiten Zuordnung der Elemente der Abbildung zur Struktur Die Ergebnisse werden in HTML-Tabellen dokumentiert	Schülerpartnerarbeit HTML-Seite <sup>16</sup>
Zwischensicherung	Vorstellung der Ergebnisse	Schüler/Beamer
Problem	Attribute und Funktionen der Klassen	
Erarbeitung	Ergänzung der Klassentabellen um mögliche Attribute und Funktionen - je Klasse ein Attribut und eine Funktion	Schülerpartnerarbeit
Zusammenfassung	Präsentation ausgewählter Schülerlösungen	Schüler/Beamer
Ausblick/Hausaufg.	Was erwarten die Schülerinnen?	

<sup>16</sup> Arbeitsanweisung (per Beamer) Die HTML-Seite erhält den Namen: account.stiftUndCo.html

## methodisch-didaktische Entscheidungen

Es zeigte sich in Diskussionen mit den Schülerinnen, dass die Motivation im Fach Informatik sehr unterschiedlich ist: einige Schülerinnen forderten eine starke Orientierung des Unterrichts zur *Bedienung* (Keyboarding) von üblichen GUIs; einige verlangten nach einer stärkeren theoretischen Durchdringung; einige wünschen die Einbeziehung der Besprechung von Hardware (und ihrer Installation) in den Unterricht.

## B.2 Modellierung grundlegender Elemente für ein „Blinklicht“

**Einordnung:** Objektorientierte Modellierung - Kontrollstrukturen; **Nutzung der Klassen:** *BuntStift*, *Farbe*, *Hilfe*

**Reihe:** Objektorientierte Modellierung

- Verwenden gegebener Klassen
- Programmgesteuertes Zeichnen
- Strukturierung mit eigener Klasse
- Malen mit der Maus:
  - Kontrollstrukturen (Schleife mit Eingangsbedingung, einseitige Verzweigung)
- Ampel - hier zuerst Blinklicht, dann Ampel
  - Steuern einer Ampel durch die Tastatur

## Stundenlernziel

Die Schülerinnen erweitern ihre Fähigkeiten zum Explorieren und Nutzen der Klassenbibliothek zur Bearbeitung eines konkreten Problems.

### kognitive Zieldimension:

Die Schülerinnen

- *kennen die Funktion des Python-Inspektionskommandos **dir(Klasse)** als Möglichkeit, Attribute und Methoden von **Klasse** herauszufinden.*
- *dokumentieren Ergebnisse einer eigenständigen Inspektion, um sie den anderen Schülerinnen vorstellen zu können.*
- *strukturieren die für die Lösung des (kleinen) Gesamtproblems notwendigen Teilschritte.*
- *nutzen die neuen Methoden **setzeFuellMuster(1)** und **setzeFarbe(meineFarben.SCHWARZ)** der Klasse **BuntStift**, um gefüllte Flächen zu erzeugen.*
- *nutzen die Methode **warte(Millisekunden)** aus der Klasse **Hilfe**, um eine Zeitverzögerung zu realisieren.*
- *setzen eine gefundene Lösungsstruktur in ein Programm um, mit dem ein **Blinklicht** realisiert wird.*

**affektive Zieldimension:**

Die Schülerinnen

- empfinden Freude an der Realisierung einer selbst entwickelten Problemlösung und der Realisierung einer korrekt funktionierenden Implementierung.

**Geplante Verlaufsstruktur**

Phase	Inhalte	Form/Medien
Einstieg	Präsentation eines <b>Blinklichts</b>	Beamer
Problem	<b>Impuls:</b> Was wird benötigt, um das Blinklicht realisieren zu können? erwartete Antworten: <i>Farbe, Füllmuster, Warten</i>	fragend- entwickelnd
Erarbeitung	<b>Arbeitsauftrag:</b> Inspektion der Klassen <b><i>BuntStift, Farbe</i></b> und <b><i>Hilfe</i></b> mit dem Ziel, die <b>notwendigen Elemente</b> für die Realisierung eines Blinklichts herauszuarbeiten und zu dokumentieren.	Arbeitsauftrag Beamer
	<b>Hinweis:</b> Nach Einlogvorgang, Aktivieren von Jython kann mit <i>from stiftUndCo import Klassenname; dir(Klassenname)</i> der Funktionsumfang einer Klasse gefunden werden Die Ergebnisse werden mit dem Editor in einer persönlichen Datei: <i>Klassenname.Dokumentation.txt</i> zusammengefasst	Partnerarbeit  Computer/Jython
Zwischensicherung	Vorstellung der Ergebnisse	Schülerin
Problem	<b>Arbeitsauftrag:</b> Realisierung eines Blinklichts durch Beschreibung (ggf. mit Hilfe) und formales Aufschreiben als Programm Abspeichern unter dem Dateinamen: <i>Blinklicht.py</i>	Arbeitsauftrag  Beamer
Erarbeitung	Unter Zuhilfenahme bereits erarbeiteter Problemlösungen wird das Programm zur Erzeugung des Blinklichts sukzessive entwickelt und interaktiv überprüft.	Partnerarbeit Computer Jython
Zusammenfassung	Präsentation ausgewählter Schülerlösungen	Schülerinnen
Ausblick/Hausaufg.	Erstellung einer Ampellichtleiste	

**B.3 Einführung in die Arbeit mit einem webbasierten CSCW zur Kursplanung****Einordnung:****Stundenlernziel**

Die Schülerinnen verstehen den CSCW-Ansatz und nutzen BSCW, um eine Diskussion über die Arbeit im Informatikkurs zu führen, Kursinhalte und -methoden abzuwägen und Entscheidungen über den weiteren Verlauf des Kurses vorzubereiten.<sup>17</sup>

<sup>17</sup> oft wird von CSCW-Systemen gesprochen/geschrieben. Hier wird die „Kurzschreibweise“ ohne das Suffix „System“ verwendet.

**kognitive und affektive Zieldimension:**

Die Schülerinnen

- helfen sich bei der Bearbeitung ihrer Aufgaben - gerade im Zusammenhang mit dem Handling auftretende Fragen können so oftmals ohne „Eingreifen“ des Lehrers geklärt werden.

**kognitive Zieldimension:**

Die Schülerinnen

- erkennen die grundlegende Struktur von CSCW mit Hilfe der Schreibtischmetapher.
- entdecken die Informations- und damit Kontrollmöglichkeiten, die CSCW mit sich bringen, indem sie die Informationen über Objekte im BSCW lesen.
- ordnen die „Awareness“-Funktion von CSCW den entsprechenden Symbolen des BSCW zu und nutzen diese Funktion, um sich über Änderungen im Arbeitsbereich (Ordner) zu informieren.
- wissen, dass mit Hilfe der Elemente in der Kopfleiste des BSCW dem Ordner neue Objekte zugeordnet werden.
- aktivieren den Verweis zu einer Notiz, lesen sie und antworten, indem sie eigene Notizen verfassen.
- erstellen Notizen, um eine „Schreibdiskussion“ über ihre Wünsche zur Ausgestaltung der Arbeit im 12. Jahrgang im Fach Informatik zu führen.

**affektive Zieldimension:**

Die Schülerinnen

- empfinden die „Schreibdiskussion“ als neue und überraschende Möglichkeit, sich über inhaltliche Fragen auseinanderzusetzen.
- spüren eine Befriedigung ihres Kommunikationsbedürfnisses durch die Nutzung des BSCW.

**psychomotorische Zieldimension:**

Die Schülerinnen

- üben ihre Fertigkeiten im Umgang mit Maus und Tastatur im Zusammenhang mit einer Applikation, die über Webseiten bedient wird.

**Geplante Verlaufsstruktur<sup>18</sup>**

Phase	Inhalte	Form/Medien
Einstieg	Abfragen und Kontrolle der Hausaufgaben: Einloggen in die Client-Server-Struktur; Einloggen in den BSCW-Server	Computer/ Einzel- und Partnerarbeit
Problem	Präsentation der Folie zu CSCW Impuls: „Schreibtisch“ „gemeinsam arbeiten“	Folie (Grafik des Arbeitsblatts ohne Text)
Erarbeitung	Schreibtischmetapher CSCW: Symbole und ihre Bedeutung Zuordnung der Elemente der Folie zur Struktur	fragend-entwickelnd Arbeitsblatt
Zwischensicherung	Vorstellung der Ergebnisse	Folie wird ausgefüllt
Problem	BSCW - die Realisierung eines webbasierten CSCW Hinweis auf „i“; Awareness; Notiz Die Kursplanung für 12.1 soll „diskutiert“ werden	Lehrerinformation ggf. Beamer (Icons) Notiz im BSCW
2. Erarbeitung	Jede Schülerin kommentiert die vorliegende Notiz. Nach Aufforderung werden eigene Notizen verfaßt	interaktive Diskussion
Zusammenfassung	im direkten Austausch werden die Erfahrungen und die Methoden einer Beurteilung durch die Schülerinnen unterzogen	Kreisgespräch
Ausblick	Die Schülerinnen fassen ihre Anmerkungen mit Hilfe weiterer Notizen im BSCW zusammen wird in der folgenden Unterrichtsstunde fortgesetzt	Notizen im BSCW Einzel-/Partnerarbeit

<sup>18</sup>Wird die *Schreibdiskussion* „gut angenommen“, ist das Kreisgespräch nicht notwendig.  
Dann wird die Zusammenfassung mit den Mitteln des BSCW durchgeführt.

## Arbeitsblatt/Folie

Diese Seite:  
 ~\UE23.06.98\Arbeitsblatt1.Loesung.html  
 eingerichtet am 20. Juni 1998

Ludger Humbert  
 letzte Änderungen: 22. Juni 1998

## CSCW

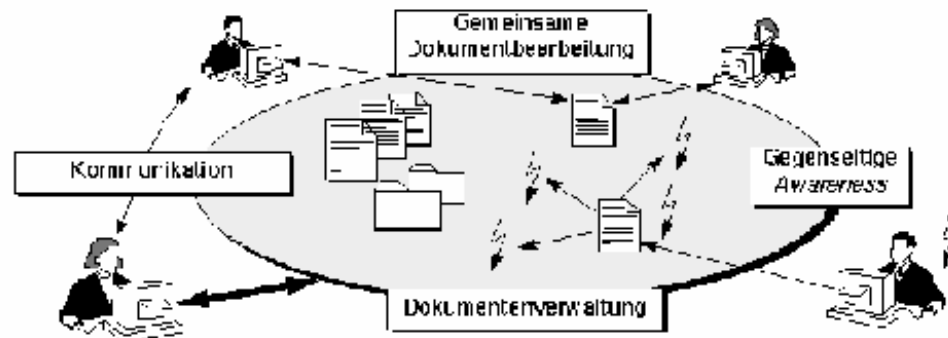







Abb.: Schematische Abbildung eines CSCW (CSCW heißt: Computer Supported Cooperative Work)

**Aufgabe: Beschreibe die einzelnen Elemente, die sich auf dem "Schreibtisch" befinden und die Bedeutung der verschiedenen Pfeiltypen:**

Symbol	Kurzbeschreibung (Stichworte)
	bezeichnet einen <b>Ordner</b> , in dem z.B. Dokumente enthalten sind
	Symbol für ein <b>Dokument</b>
	Mitteilung, Weitergeben eines Dokuments - Aufmerksamkeit erwecken: <b>Awareness</b>
	"dicke" beidseitige <b>Verbindung</b> zu dem CSCW
	"dünne" beidseitige <b>Verbindung</b> zwischen Elementen



## B.4 Das Fachkonzept Keller - Realisierung als Klasse

### Einordnung:

- **Reihe:** Automaten - Transduktoren, Akzeptoren
- **Einheit:** Graphische Darstellungen in der Informatik
- **Reihe:** Automatische Erstellung strukturierter Dokumente - informatische Fachkonzepte
  - Doppelstunde: Vorüberlegungen zur Klammerstruktur in XML Dokumenten am Beispiel von HTML
  - Einzelstunde: die Klasse Keller (Modellierungsüberlegungen)
  - Einzelstunde: Das Fachkonzept Keller - Vorüberlegungen zur Umsetzung mit Hilfe der Datenstruktur Liste
  - **Doppelstunde:** Das Fachkonzept Keller - Realisierung als Klasse

### Stundenlernziel

Die Schülerinnen implementieren die Klasse **Keller** durch Benutzen der Datenstruktur Liste.

### kognitive Zieldimension:

Die Schülerinnen

- erkennen, dass die Methode `__init__` (Konstruktor) eine leere Liste erzeugen muss.
- erkennen, dass die Methode `push(element)` durch die Listenmethode `append` realisiert werden kann
- erkennen, dass die Methode `pop()` durch zwei Aktionen auf der Liste realisiert wird: Zurückliefern des letzten Listenelements und Löschen des letzten Elements.
- erkennen, dass die Methode `empty()` die Anzahl der Elemente prüfen muss.
- erstellen die Klasse Keller in dem sie die Methoden `__init__`, `push`, `pop` und `empty` füllen.
- entwickeln eine Testmöglichkeit für die erstellte Klasse Keller und fügen diese dem Quellcode hinzu. (fakultativ)

### affektive Zieldimension:

Die Schülerinnen

- stellen erfreut fest, dass die Klasse **Keller** kompakt implementiert werden kann und damit eine einfache Möglichkeit bereitsteht, die geforderte Funktionalität zur Verfügung zu stellen.

**Geplante Verlaufsstruktur**

Phase	Inhalte	Form/Medien
Einstieg	Zusammenstellung der Anforderungen für die Klasse Keller bisherige Erkenntnisse zur Umsetzung Zusammenhänge zwischen Liste und Keller	Lehrerimpuls  Schülerantworten Tafel
Problem	konkrete Realisierung der Klasse Keller	
Erarbeitung	formaler Rahmen: class ... def ...	fragend-entwickelnd/Heft
Zwischensicherung	Vorstellung ausgewählter Ergebnisse	Schülerinnen: Heft/Tafel
Erarbeitung	Eingabe der Klasse und Prüfung der syntaktischen Korrektheit	Schüler/Computer Editor und Jython
Problem	Testroutine für alle Methoden der Klasse	Lehrerimpuls
Erarbeitung	Entwicklung und Realisierung einer Testroutine Was muss in welcher Reihenfolge getestet werden? Entscheidung und Umsetzung Instanziierung der Klasse Keller, Prüfung mit empty pop bei leerem Keller? mehrmaliges push und dann pop	fragend-entwickelnd Schüler/Computer Editor und Jython  Vorführung durch Schülerinnen/Beamer
Ausblick	Erweiterung: Einrücken	Zusammenfassung und Diskussion

**Tafel**

Zum Zusammenhang zwischen Keller und Liste

Wie kann die Klasse Keller mit einer Liste realisiert werden?

<b>Klasse Keller</b>	
Attribut	
lifo	Liste
Methoden	
<code>__init__(self)</code>	(Konstruktor)
<code>push(self, element)</code>	
<code>pop(self)</code>	element
<code>empty(self)</code>	true oder false

<b>Datenstruktur Liste</b>
<code>self.lifo=[ ]</code>
<code>self.lifo.append(element)</code>
<code>element=self.lifo[-1]</code>
<code>del (self.lifo[-1])</code>
<code>return element</code>
<code>return len(self.lifo)==0</code>

---

## Material

```
class Keller:
    'Implementierung eines Kellers als Liste'

    def __init__(self):
        '''vorher: -,
        nachher: eine leere Liste, die den Stapel aufnehmen soll, ist angelegt
        '''
        self.lifo=[]

    def push(self,element):
        '''vorher: es gibt einen Stapel,
        nachher: element liegt oben auf dem Stapel'''
        self.lifo.append(element)

    def pop(self):
        '''liefert das oberste Element eines Stapels zurück;
        vorher: Stapel,
        nachher: element ist vom Stapel entfernt'''
        if not self.empty():
            element=self.lifo[-1]
            del self.lifo[-1]
        else:
            print 'der Stapel/Keller ist leer; pop ist nicht möglich'
            element=None
        return element

    def empty(self):
        '''liefert zurück, ob der Stapel leer ist
        vorher Stapel, nachher Stapel (der Stapel wird nicht verändert)'''
        return len(self.lifo)==0

    def top(self):
        '''das obere Element des 'Stapels' wird zurückgegeben (ansehen)'''
        return self.lifo[-1]

if __name__ == '__main__':
    ersterKeller = Keller()
    zweiterKeller = Keller()
    if ersterKeller.empty(): print 'ersterKeller ist leer'
    ersterKeller.push('</html>')
    # ...
    ersterKeller.push('</table>')
    print 'letztes Element auf ersterKeller:', ersterKeller.pop()
    zweiterKeller.push(ersterKeller.pop())
    print 'letztes Element auf zweiterKeller:', zweiterKeller.pop()
    if zweiterKeller.empty(): print 'zweiterKeller leer'
    while not ersterKeller.empty():
        print ersterKeller.pop()
    if ersterKeller.empty(): print 'ersterKeller leer'
    ersterKeller.pop()
```

## B.5 Nebenläufigkeit zur Realisierung der „Parallelität“ von Server und Klient

### Einordnung:

**Reihe:** Intranet/Internet aus Sicht von Informatiksystemen

- Schichtenmodell
- Vorstellung der Klasse UDFsocket (UDFsocket.py)
  - Realisierung von Klienten
- Realisierung von Servern
- Nebenläufigkeit

### Stundenlernziel

Die Schülerinnen beschreiben Anforderungen an die Struktur der parallelen Nutzung der Funktionalität von Server- und Klientenaktivitäten zum Nachrichtenaustausch im schulischen Intranet und realisieren prototypische Lösungen durch die Einrichtung eines nebenläufigen Prozesses.

### kognitive Zieldimension:

Die Schülerinnen

- *kennen und nutzen die Begriffe: Klient, Server, Prozess.*
- *erkennen, dass das Problem, gleichzeitig Server (zur Entgegennahme von Daten) und Klient (der aktiv Daten aussenden soll) durch ein Programm realisieren zu wollen, sich nur dadurch lösen lässt, dass ein (neuer) Server-Prozess erzeugt wird, der im Hintergrund arbeitet.*
- *beschreiben die grundlegende Idee der Nebenläufigkeit mit eigenen Worten.*
- *wissen, dass **Nebenläufigkeit** programmtechnisch durch die Einrichtung von sog. **Threads** umgesetzt werden kann.*
- *realisieren die Pseudoparallelität von Server- und Klientenprozess durch die Installation eines nebenläufigen Prozesses, der die Serverfunktion realisiert.*

## Geplante Verlaufsstruktur

Phase	Inhalte	Form/Medien
Einstieg	aktuelle Lösungen - Server und Klient getrennte Prozesse	Schülerinnen Intranet, Beamer
Problem	gleichzeitig Server und Klient in einem Skript realisieren	
Erarbeitung	Ideensammlung Parallelität rote <b>Fäden</b> (Threads)	Schülerinnen  Lehrerinformation
Zwischensicherung	Definitionen der Fachbegriffe im Kontext	Folie/Beamer
Problem	Umsetzung der Nebenläufigkeit Python-Spezifika zur Parametrisierung von <i>start_new_thread</i>	Lehrererklärung - Beamer <i>from thread import start_new_thread</i>
2. Erarbeitung	Realisierung - Umsetzung der Lösungsideen für den speziellen Fall	Schülerinnen, Informatiksysteme Intranet
Zusammenfassung	Vorstellung von Lösungsansätzen und Thematisieren von Problemen bei der Umsetzung	Schülerinnen
Ausblick	Planung Anforderungen an erweiterte Problemstellungen / -lösungen	

Folie/Tafel<sup>19</sup>

Marco Thomas - Nebenläufigkeit im Informatikunterricht - Universität Potsdam - Didaktik der Informatik - Oberhausen, 18.09.1998

2

## Nebenläufigkeit

### Def. 1

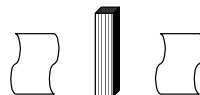
Zwei Ereignisse/Vorgänge heißen **nebenläufig**, wenn sie

- zeitweilig voneinander unabhängig auftreten bzw. ablaufen können und
- abhängigkeiterzeugende Wirkungszusammenhänge zwischen ihnen bestehen können.



### Def. 2

Nebenläufige Ereignisse oder Vorgänge heißen **parallel** oder gleichzeitig, wenn zwischen ihnen keine Wirkungszusammenhänge bestehen, die die Unabhängigkeit beeinflussen.



### Beispiele

- Während eines Vortrags können die Zuhörer parallel irgendwelchen anderen Tätigkeiten nachgehen.
- Ein zum Vortrag nebenläufiges Gespräch zwischen zwei Zuhörern zu einer aufgelegten Vortragsfolie kann gestört werden, wenn eine andere Vortragsfolie auf den Projektor aufgelegt wird.
- Zwei Prozesse auf einem Rechner mit einer CPU können nebenläufig, aber nicht parallel ablaufen.
- Zwei nicht-vernetzte Rechner arbeiten vollständig parallel (gleichzeitig).

## methodisch-didaktische Entscheidungen

Nebenläufigkeit ist als eines der Themen anzusehen, denen ausgewiesenermaßen eine hohe Bedeutung in der Arbeit mit verteilten Systemen zukommt. Vor allem in der Erstellung von Skripten, die, wie in dem Fall des Servers „permanent warten“ stellt sich die Frage, wie eine Umsetzung so erfolgen kann, dass mit dem System während dieser Wartezeit auch noch andere Arbeiten erledigt werden können.

In diesem Fall bietet sich Nebenläufigkeit als Lösungsidee geradezu an.

Zur Umsetzung kann in der hier zur Verwendung gebrachten Programmiersprache Jython auf ein mächtiges Sprachkonzept zurückgegriffen werden, das es ermöglicht, eine Funktion durch die Aktivierung einer Funktion (`thread.start_new_thread`) zur Nebenläufigkeit zu veranlassen.

<sup>19</sup>THOMAS, Marco Nebenläufigkeit im Schulfach Informatik - Legitimation und Vermittlung. <http://didaktik.cs.uni-potsdam.de/HyFISCH/Informieren/Nebenlaeufigkeit/Nebenlaeufigkeit.pdf>. September 1998, Folie 2

---

## Material

```

from UDFsocket import UDFsocket
from thread import start_new_thread

class Server:
    def __init__(self, art="Server", meinName="ada", meinPort=8081):
        self.port=meinPort
        self.serverSteckdose= UDFsocket(art, meinName, self.port)

    def warteAufDaten(self):
        def warteSchleife(self):
            i=1
            while 1:
                eingangsDaten= self.serverSteckdose.datenempfang()
                print "\nEMPfang:", i, eingangsDaten, "von: Rechner", \
                    self.serverSteckdose.getAbsender()[0], \
                    "Port:", self.serverSteckdose.getAbsender()[1]
                i=i+1
            print "Server wartet auf Verbindung am Port:", self.port
            start_new_thread(warteSchleife, (self, ))

class Klient:
    def __init__(self, art="Klient", ServerName="ada", meinPort=8081):
        self.sender=UDFsocket(art, ServerName, 8081)

    def sende(self, text="Holy Guido! It's working."):
        self.sender.datenaussenden(text)

    def kontinuierlichTastatureingabensenden(self):
        while 1:
            naechsterText=raw_input("Was soll gesendet werden? ")
            self.sender.datenaussenden(naechsterText)
            print "... wird gesendet ..."

#-----
if __name__=="__main__":
    meinServer=Server(meinName="babagge")
    # der Server soll auf ankommende Daten warten - ...
    meinServer.warteAufDaten()

    meinKlient=Klient(ServerName="babagge")
    meinKlient.sende("mein Klient sendet mit freundlichen Grüßen ...")

    meinKlient.kontinuierlichTastatureingabensenden()

```

---

```
import socket

# Standardeinstellungen

class UDFsocket:
    def __init__(self, art="Klient", server="localhost", port=8081):
        self.rechnername=socket.gethostname()
        self.art= art
        self.server= server
        self.port= port
        self.absenderAdresse=server
        self.sockel = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        if art=="Server":
            self.sockel.bind((self.server, self.port))
        else: self.sockel.bind(("", 0))

    def datenempfang(self):
        daten, self.absenderAdresse = self.sockel.recvfrom(1024)
        return daten

    def getAbsender(self):
        return self.absenderAdresse

    def datenaussenden(self, daten):
        self.sockel.sendto(daten, (self.server, self.port))

    def dieserRechner(self):
        return self.rechnername
```



---

```

from UDFsocket import UDFsocket
from thread import start_new_thread
from java import awt
from pawt import swing

class Server:
    def __init__(self, art="Server", meinName="", meinPort=8081):
        self.port=meinPort
        self.serverSteckdose= UDFsocket(art, meinName, self.port)
        self.empfangeneDaten=""

    def warteAufDaten(self):
        def warteSchleife(self):
            i=1
            while 1:
                eingangsDaten= self.serverSteckdose.datenempfang()
                self.empfangeneDaten=str(i)+ " " +str(eingangsDaten) + \
                " von: Rechner " + str(self.serverSteckdose.getAbsender()[0]) + \
                " Port: " + str(self.serverSteckdose.getAbsender()[1])
                self.datenanzeige(self.empfangeneDaten)
                i=i+1
            self.empfangeneDaten="Server wartet auf Verbindung am Port:"+str(self.port)
            start_new_thread(warteSchleife, (self, ))

    def datenanzeige(self, text):
        pass # die Methode muss überschrieben werden

class Klient:
    def __init__(self, art="Klient", ServerName="", meinPort=8081):
        self.sender=UDFsocket(art, ServerName, 8081)

    def sende(self, text="Holy Guido! It's working."):
        self.sender.datenaussenden(text)
        print "Klient:",text

class Modell:
    def __init__(self,rechnername="haspe", servername="haspe"):
        self.meinServer=Server(meinName=rechnername) # hat ein Objekt der Klasse Server
        self.meinServer.warteAufDaten()
        self.meinKlient=Klient(ServerName=servername) # hat ein Objekt der Klasse Klient
        self.meinKlient.sende(" Na ... es klappt ...")

class ViewControl:
    def __init__(self, modell):
        self.modell=modell # kennt das Modell
        self.aufschrift = 'eingegebene Daten absenden'
        self.knoepfe = swing.JPanel(awt.GridLayout(1, 1)) # Anordnung
        self.serverDatenempfang = swing.JTextField()
        self.modell.meinServer.datenanzeige = self.serverDatenempfang.setText # Methode füllen
        self.klientenDaten = swing.JTextField()
        self.knopf = swing.JButton(self.aufschrift)
        self.knopf.actionPerformed = self.aktionDruecke
        oberflaeche = swing.JPanel(awt.BorderLayout())
        oberflaeche.add("North", self.serverDatenempfang)
        oberflaeche.add("Center", self.klientenDaten)
        oberflaeche.add("South", self.knopf)
        swing.test(oberflaeche)

    def aktionDruecke(self,ereignis): # Verbindung von Ereignis und Aktion
        self.modell.meinKlient.sende(self.klientenDaten.text)
        self.klientenDaten.setText("")

if __name__=="__main__":
    modell= Modell("stec", "stec")
    meinMVC= ViewControl(modell)

```

## Literaturverzeichnis

- [Appelt und Busbach 1996] APPELT, Wolfgang ; BUSBACH, Uwe: The BSCW System: A WWW based Application to Support Cooperation of Distributed Groups. In: *Proc. of WET ICE 96: Collaborating on the Internet: The World-Wide Web and Beyond*. Los Alamitos : IEEE Computer Society Press, June 1996, S. 304–310
- [Arlt 1981] ARLT, Wolfgang (Hrsg.): *Informatik als Schulfach. Didaktische Handreichungen für das Schulfach Informatik*. München : Oldenbourg, 1981 (Datenverarbeitung im Bildungsbereich 1)
- [Arlt u. a. 1982] ARLT, Wolfgang ; DIRNBERGER, Josef ; KLÖCKNER, Konrad ; SCHAUER, Helmut ; SCHÜTZ, Ute ; TAVOLATO, Paul ; HAAS, Hans W. (Hrsg.) ; WILDENBERG, Detlef (Hrsg.): *Informatik für Lehrer Studentexte und Handreichungen für den Unterricht*. Bd. 1 : Einführung in die Schulinformatik. München : Oldenbourg Verlag, 1982
- [Association for Computing Machinery 1993] ASSOCIATION FOR COMPUTING MACHINERY: *Model High School Computer Science Curriculum*. New York : ACM (Association for Computing Machinery), 1993. – Siehe auch <http://www.acm.org/education/hscur/index.html>
- [Association for Computing Machinery 1997] ASSOCIATION FOR COMPUTING MACHINERY: *Model High School Computer Science Curriculum*. <http://www.acm.org/education/hscur/index.html>. November 1997. – HTML-Version
- [Balzert 1977] BALZERT, Helmut: *Informatik: 1. Vom Problem zum Programm – Lösungsband mit methodisch-didaktischer Einführung*. 1. Aufl. München : Hueber-Holzmann Verlag, 1977
- [Balzert 1979] BALZERT, Helmut: *Informatik: 1. Vom Problem zum Programm – Lösungsband mit methodisch-didaktischer Einführung*. 2. Aufl. München : Hueber-Holzmann Verlag, 1979. – 1. Aufl. 1977
- [Bauersfeld u. a. 1977] BAUERSFELD, H. (Hrsg.) ; OTTE, M. (Hrsg.) ; STEINER, Hans G. (Hrsg.): *Informatik im Unterricht der Sekundarstufe II: Grundfragen, Probleme und Tendenzen mit Bezug auf allgemeinbildende und berufsqualifizierende Ausbildungsgänge*. Bielefeld : Universität Bielefeld, 1977 (Schriftenreihe des IDM (Institut für Didaktik der Mathematik) Nr 15 (Band I) und 16 (Band II)). – Arbeitstagung: Bielefeld 12.-14. September 1977
- [Baumann 1993] BAUMANN, Rüdiger: Ziele und Inhalte des Informatikunterrichts. In: *Zentralblatt für Didaktik der Mathematik* 25 (1993), Nr. 1, S. 9–19. – <http://ddi.cs.uni-dortmund.de:8080/lehre/hauptstudium/ws2001/uebung/baumann.pdf>
- [Berger 1997] BERGER, Peter: Das 'Computer-Weltbild' von Lehrern. In: **[Hoppe und Luther 1997]**, S. 27–39
- [BLK 1984] BLK (Hrsg.): *Rahmenkonzept für die Informationstechnische Bildung in Schule und Ausbildung*. Bonn : Bund-Länder-Kommission für Bildungsplanung und Forschungsförderung (BLK), 1984
- [Brauer und Brauer 1992] BRAUER, Wilfried ; BRAUER, Ute: Wissenschaftliche Herausforderungen für die Informatik: Änderung von Forschungszielen und Denkgewohnheiten. In: LANGENHEDER, Werner (Hrsg.) ; MÜLLER, G. (Hrsg.) ; SCHINZEL, Britta (Hrsg.): *Informatik cui bono?* Berlin : Springer, 1992 (Informatik aktuell Bd. 15), S. 11–19
- [Brauer und Brauer 1995] BRAUER, Wilfried ; BRAUER, Ute: Informatik – das neue Paradigma (Änderung von Forschungszielen und Denkgewohnheiten der Informatik). In: *LOG IN* 15 (1995), Nr. 4, S. 25–29
- [Brauer u. a. 1976] BRAUER, Wilfried ; CLAUS, Volker ; DEUSSEN, Peter ; JÜRGEN EICKEL (FEDERFÜHREND) ; HAACKE, Wolfhart ; HOSSEUS, Winfried ; KOSTER, Cornelis H. A. ; OLLESKY, Dieter ; WEINHART, Karl ; GESELLSCHAFT FÜR INFORMATIK E. V.: Zielsetzungen und Inhalte des Informatikunterrichts. In: *Zentralblatt für Didaktik der Mathematik, Stuttgart* 8 (1976), Nr. 1, S. 35–43

- [Brooks 1986] BROOKS, Frederick P.: No Silver Bullet – Essence and Accidents of Software Engineering. In: **[Kugler 1986]**,
- [Brooks 1987] BROOKS, Frederick P.: No Silver Bullet – Essence and Accidents of Software Engineering. In: *IEEE Computer* 20 (1987), Nr. 1, S. 10–19. – <http://www.student.math.uwaterloo.ca/~cs212/resource/Articles/SilverBullet.html>
- [Buse 1980] BUSE, Dirk: Informatik. In: ROTH, Leo (Hrsg.): *Handlexikon der Didaktik der Schulfächer*. München : Ehrenwirth, 1980, S. 256–264
- [Capurro 1990] CAPURRO, Rafael: Ethik und Informatik. In: *Informatik Spektrum* 13 (1990), Dezember, Nr. 6, S. 311–320
- [Claus 1975] CLAUS, Volker: *Einführung in die Informatik*. Stuttgart : Teubner, 1975
- [Claus 1977] CLAUS, Volker: Informatik an der Schule: Begründungen und allgemeinbildender Kern. In: **[Bauersfeld u. a. 1977]**, S. 19–33. – Band I
- [Claus u. a. 1979] CLAUS, Volker ; EICKEL, Jürgen ; GUNZENHÄUSER, Rul ; HACKL, Clemens ; HOSSEUS, Winfried ; LOFF, Jürgen ; SCHAUER, Helmut ; SCHELL-HAUNGS, Isolde ; SCHULZ-ZANDER, Renate ; SPITTA, Gerhard: Empfehlungen zur Ausbildung, Fortbildung und Weiterbildung von Lehrkräften für das Lehramt Informatik für die Sekundarstufe I und II. In: *Informatik Spektrum* 2 (1979), Februar, Nr. 1, S. 53–60
- [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)
- [Denning 1999a] DENNING, Peter J.: Editor's introductory essay: Computers and Human Aspiration. In: *Talking back to the Machine: Computers and Human Aspiration***[Denning 1999b]**, S. xi–xviii. – <http://cne.gmu.edu/pjd/PUBS/thennow99.pdf>
- [Denning 1999b] DENNING, Peter J. (Hrsg.): *Talking back to the Machine: Computers and Human Aspiration*. New York : Copernicus Books (Springer), May 1999b
- [Denning und Metcalfe 1997] DENNING, Peter J. (Hrsg.) ; METCALFE, Robert M. (Hrsg.) ; ACM (Veranst.): *Beyond Calculation: The Next 50 Years of Computing*. New York : Copernicus Books (Springer), May 1997 . – First softcover printing 1998
- [Dostal 2000] DOSTAL, Werner: Die Informatisierung der Arbeitswelt - Ein erster Blick auf die Ergebnisse der BIBB/IAB-Erhebung. In: DOSTAL, Werner (Hrsg.) ; JANSEN, Rolf (Hrsg.) ; PARMENTIER, Klaus (Hrsg.): *Wandel der Erwerbsarbeit: Arbeitssituation, Informatisierung, berufliche Mobilität und Weiterbildung*. Nürnberg : Bundesanstalt für Arbeit, 2000, S. 151–167
- [Eberle 1996] EBERLE, Franz ; WETTSTEIN, Emil (Hrsg.) ; WEIBEL, Walter (Hrsg.) ; GONON, Philipp (Hrsg.): *Pädagogik bei Sauerländer: Dokumentation und Materialien*. Bd. 24: *Didaktik der Informatik bzw. einer informations- und kommunikationstechnologischen Bildung auf der Sekundarstufe II – Ziele und Inhalte, Bezug zu anderen Fächern sowie unterrichtspraktische Handlungsempfehlungen*. 1. Aufl. Aarau : Verlag Sauerländer, 1996
- [Engbring 1995] ENGBRING, Dieter: Kultur- und technikgeschichtlich begründete Bildungswerte der Informatik. In: **[Schubert 1995]**, S. 68–77

- [Fischer u. a. 2000] FISCHER, Arthur ; FRITZSCHE, Yvonne ; FUCHS–HEINRITZ, Werner ; MÜNCHMEIER, Richard: *Jugend 2000 – 13. Shell Jugendstudie (2 Bände)*. Opladen : Leske + Budrich, 2000
- [Fleischhut u. a. 1982] FLEISCHHUT, Jens ; DIRNBERGER, Josef ; KOERBER, Bernhard ; REKER, Jörg ; RIEDEL, Dieter ; SCHAUER, Helmut ; STEPHAN, Christine ; TAVOLATO, Paul ; HAAS, Hans W. (Hrsg.) ; WILDENBERG, Detlef (Hrsg.): *Informatik für Lehrer Studentexte und Handreichungen für den Unterricht*. Bd. 2 : Komplexere Probleme und Didaktik der Schulinformatik. München : Oldenbourg Verlag, 1982
- [Floyd 1992] FLOYD, Christiane: Human Questions in Computer Science. In: FLOYD, Christiane (Hrsg.) ; ZÜLLIGHOVEN, Heinz (Hrsg.) ; BUDDE, Reinhard (Hrsg.) ; KEIL-SLAWIK, Reinhard (Hrsg.): *Software Development and Reality Construction*. Berlin : Springer, 1992, S. 15–27
- [Friedrich 1995] FRIEDRICH, Steffen: Standortbestimmung der Informatik in der Schule. In: **[Schubert 1995]**, S. 33–39
- [Genrich 1975] GENRICH, Hartmann J.: Belästigung des Menschen durch Computer. In: *Der GMD–Spiegel* (1975), Dezember, Nr. 5, S. 32–45
- [Gesellschaft für Informatik 2000] GESELLSCHAFT FÜR INFORMATIK: Empfehlung der Gesellschaft für Informatik e.V. für ein Gesamtkonzept zur informatischen Bildung an allgemein bildenden Schulen. In: *Informatik Spektrum* 23 (2000), Dezember, Nr. 6, S. 378–382. – siehe auch: [http://ddi.cs.uni-dortmund.de/ddi\\_bib/gi\\_empfehlung/gesamt2000/gesamtkonzept-26-9-2000.pdf](http://ddi.cs.uni-dortmund.de/ddi_bib/gi_empfehlung/gesamt2000/gesamtkonzept-26-9-2000.pdf)
- [GfK Marktforschung GmbH u. a. 2000] GfK MARKTFORSCHUNG GMBH ; FRAUNHOFER-INSTITUT FÜR EXPERIMENTELLES SOFTWARE ENGINEERING IESE ; FRAUNHOFER-INSTITUT FÜR SYSTEMTECHNIK UND INNOVATIONSFORSCHUNG ISI: *Analyse und Evaluation der Softwareentwicklung in Deutschland*. [http://www.dlr.de/IT/IV/Studien/evasoft\\_abschlussbericht.pdf](http://www.dlr.de/IT/IV/Studien/evasoft_abschlussbericht.pdf). Dezember 2000. – Studie für das Bundesministerium für Bildung und Forschung
- [Goos 1979] GOOS, Gerhard: Editorial Informatik an der Schule? In: *Informatik Spektrum* 2 (1979), Februar, Nr. 1, S. 1–3
- [Häußler u. a. 1998] HÄUSSLER, Peter (Hrsg.) ; BÜNDER, Wolfgang (Hrsg.) ; DUIT, Reinders (Hrsg.) ; GRÄBER, Wolfgang (Hrsg.) ; MAYER, Jürgen (Hrsg.): *Naturwissenschaftsdidaktische Forschung - Perspektiven für die Unterrichtspraxis*. Kiel : Institut für die Pädagogik der Naturwissenschaften, 1998
- [Håndlykken und Nygaard 1981] HÅNDLYKKEN, P. ; NYGAARD, Kristen: The DELTA System Description Language – Motivation, Main Concepts and Experience from Use. In: HÜNKE, H. (Hrsg.): *Software Engineering Environments Proceedings of the Symposium held in Lahnstein, Germany, June 16-20, 1980*. Amsterdam : North-Holland, 1981
- [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 1997] HUBWIESER, Peter: Wie soll der Unterricht ablaufen? In: **[Hoppe und Luther 1997]**, S. 216
- [Hubwieser und Broy 1996] HUBWIESER, Peter ; BROY, Manfred: Der informationszentrierte Ansatz - Ein Vorschlag für eine zeitgemäße Form des Informatikunterrichtes am Gymnasium / Technische Universität München - Fakultät für Informatik. München, Mai 1996 ( TUM-I9624). – Forschungsbericht

- [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 1999] HUMBERT, Ludger: Grundkonzepte der Informatik und ihre Umsetzung im Informatikunterricht. In: [Schwill 1999], S. 175–189
- [Humbert 2001] HUMBERT, Ludger: Informatik lehren - zeitgemäße Ansätze zur nachhaltigen Qualifikation aller Schülerinnen. In: KEIL-SLAWIK, Reinhard (Hrsg.) ; MAGENHEIM, Johannes (Hrsg.): *Informatik und Schule - Informatikunterricht und Medienbildung*. Bonn : Gesellschaft für Informatik, Köllen Druck + Verlag GmbH, September 2001 (GI-Edition – Lecture Notes in Informatics – Proceedings P-8), S. 121–132
- [Humbert u. a. 2000] HUMBERT, Ludger ; STREITBERG, Sanna ; RUX, Martina ; DANIČIČ, Josef ; EMONTS-GAST, Martin ; GRUBERT, Volker: *Grundlegende Unterrichtskonzepte der Informatik und ihre Umsetzung in der zweiten Phase der Lehrerinnenausbildung. Zur Verzahnung von Theorie und Praxis*. <http://didaktik.cs.uni-potsdam.de/HyFISCH/WorkshopLehrerbildung2000/Papers/Humbert.pdf>. Juli 2000. – Beitrag zum Workshop zur Lehrerbildung, GI-Jahrestagung 2000, Berlin, 19. September 2000
- [Jochum 1998] JOCHUM, Heiko: *Objektorientierung zur Analyse, zum Design und zur Programmierung am Beispiel eines Strategiespiels mit einem Schwerpunkt in arbeitsteiliger Gruppenarbeit*. 1998. – 2. Staatsarbeit für das Lehramt für die Sekundarstufe II, Informatik <http://bscw.hagen.de/pub/german.cgi/d204238/>
- [Klafki 1985] KLAFKI, Wolfgang: Konturen eines neuen Allgemeinbildungskonzepts. In: *Neue Studien zur Bildungstheorie und Didaktik: Beiträge zur kritisch-konstruktiven Didaktik*. Weinheim, Basel : Beltz Verlag, 1985, S. 12–30
- [Knöß 1989] KNÖSS, Petra: *Fundamentale Ideen der Informatik im Mathematikunterricht*. Wiesbaden : Deutscher Universitäts-Verlag, 1989
- [Krämer 1997] KRÄMER, Sybille: Werkzeug – Denkzeug – Spielzeug. Zehn Thesen über unseren Umgang mit Computern. In: [Hoppe und Luther 1997], S. 7–13
- [Kubicek 1996] KUBICEK, Herbert: Die Rolle der Schulen auf dem Weg in die Informationsgesellschaft, Vortrag 6, Tagung des Bundesarbeitskreises Netze in Schulen. In: *Material für die Teilnehmer* (1996). – <http://schulen.hagen.de/IF/IN/Ueberblick.html>
- [Kugler 1986] KUGLER, H.-J. (Hrsg.) ; IFIP (Veranst.): *Information Processing 86*. North-Holland : Elsevier Science Publishers B.V., September 1986
- [Kultusministerium des Landes Nordrhein-Westfalen 1993] KULTUSMINISTERIUM DES LANDES NORDRHEIN-WESTFALEN (Hrsg.): *Richtlinien und Lehrpläne Informatik Gymnasium Sekundarstufe I*. Frechen : Verlagsgesellschaft Ritterbach, April 1993 (Die Schule in Nordrhein-Westfalen)
- [Kultusministerium Mecklenburg-Vorpommern 1998] KULTUSMINISTERIUM MECKLENBURG-VORPOMMERN (Hrsg.): *Rahmenplan Informatische Bildung für die allgemein bildenden Schulen in Mecklenburg-Vorpommern*. Schwerin : KM M-V, 1998. – <http://www.bildung-mv.de/download/rahmenplaene/rp-informatische-grundbildung-5-6.pdf>
- [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

- [Lockemann 1986] LOCKEMANN, Peter C.: Konsistenz, Konkurrenz, Persistenz – Grundbegriffe der Informatik? In: *Informatik Spektrum* 9 (1986), Oktober, Nr. 5, S. 300–305
- [Luft 1992] LUFT, Alfred L.: Grundlagen einer Theorie der Informatik: «Wissen» und «Information» bei einer Sichtweise der Informatik als Wissenstechnik. In: [Coy u. a. 1992], S. 49–70
- [Magenheim u. a. 1999] MAGENHEIM, Johannes ; SCHULTE, Carsten ; HAMPEL, Thorsten: Dekonstruktion von Informatiksystemen als Unterrichtsmethode – Zugang zu objektorientierten Sichtweisen im Informatikunterricht. In: [Schwill 1999], S. 149–164
- [Meyer 1988] MEYER, Hilbert: *Unterrichtsmethoden*. Bd. I: Theorieband. 2. Aufl. Frankfurt a. M. : Scriptor Verlag, 1988
- [Meyer 1989] MEYER, Hilbert: *Unterrichtsmethoden*. Bd. II: Praxisband. 2. durchges. Aufl. Frankfurt a. M. : Scriptor Verlag, 1989
- [Mulder und van Weert 1999] MULDER, Fred ; VAN WEERT, Tom: IFIP/UNESCO Informatics Curriculum Framework 2000. In: *IFIP Newsletter* 16 (1999), Juni, Nr. 2, S. 3–4. – ICF–2000
- [Nievergelt 1991] NIEVERGELT, Jürg: *Was ist Informatik-Didaktik?* Oldenburg : Universität, 1991. – Sonderdruck 4. Fachtagung „Informatik und Schule“
- [Nievergelt 1993] NIEVERGELT, Jürg: Was ist Informatik-Didaktik? Gedanken über die Fachkenntnisse des Informatiklehrers. In: *Informatik Spektrum* 16 (1993), Februar, Nr. 1, S. 3–10
- [Nygaard 1986] NYGAARD, Kristen: Program Development as a Social Activity. In: [Kugler 1986], S. 189–198. – [http://www.ifi.uio.no/~kristen/PDF\\_MAPPE/F\\_PDF\\_MAPPE/F\\_IFIP\\_86.pdf](http://www.ifi.uio.no/~kristen/PDF_MAPPE/F_PDF_MAPPE/F_IFIP_86.pdf)
- [Parnas 1986] PARNAS, David L.: Software Wars. Offener Brief an Mr. H. Offut, Ministerium für Verteidigung, Washington. In: *Kursbuch* 83, *Rotbuch Verlag, Berlin* (1986), März, S. 49–69. – Aus dem Amerikanischen von Philip Bacon
- [Peschke 1989] PESCHKE, Rudolf: Die Krise des Informatikunterrichts in den neunziger Jahren. In: STETTER, Franz (Hrsg.) ; BRAUER, Wilfried (Hrsg.): *Informatik und Schule 1989: Zukunftsperspektiven der Informatik für Schule und Ausbildung*. Berlin, Heidelberg : Springer, 1989 (Informatik-Fachberichte 220), S. 89–98
- [Petri 1983] PETRI, Carl A.: Zur 'Vermenschlichung' des Computers. In: *Der GMD-Spiegel* (1983), Nr. 3/4, S. 42–44
- [Rickert u. a. 2001] RICKERT, Wibke ; CREMER, Thomas ; DESCHEPPER, Patrick ; HUMBERT, Ludger: Qualifizierung von Informatiklehrern in der zweiten Ausbildungsphase der Lehrerbildung. Bonn : Gesellschaft für Informatik, Köllen Druck + Verlag GmbH, September 2001 (GI-Edition – Lecture Notes in Informatics – Proceedings P-8), S. 223–226
- [Schauer und Tauber 1983] SCHAUER, Helmut (Hrsg.) ; TAUBER, Michael J. (Hrsg.): *Psychologie des Programmierens*. Wien, München : Oldenbourg Verlag, 1983
- [Schubert 1991] SCHUBERT, Sigrid: Fachdidaktische Fragen der Schulinformatik und (un)mögliche Antworten. In: GORNY, Peter (Hrsg.): *Informatik: Wege zur Vielfalt beim Lehren und Lernen* Bd. 292. Berlin, Heidelberg : Springer, 1991, S. 27–33
- [Schubert 1995] SCHUBERT, Sigrid (Hrsg.): *Innovative Konzepte für die Ausbildung*. Berlin, Heidelberg : Springer, 1995 (Informatik aktuell)

- [Schubert und Schwill 1996] SCHUBERT, Sigrid ; SCHWILL, Andreas: Informatik in der Schule – ein Fach im Wandel. Informationen vom 3. Fachdidaktischen Gespräch zur Informatik an der Technischen Universität Dresden. In: *LOG IN* 16 (1996), Nr. 2, S. 32–33. – Ergebnisse der Arbeitsgruppe: Strukturierung des Schulfachs Informatik aus der Sicht der Fachwissenschaft
- [Schulz-Zander u. a. 1993] SCHULZ-ZANDER, Renate ; BRAUER, Wilfried ; BURKERT, Jürgen ; HEINRICHS, U. ; HILTY, Lorenz M. ; HÖLZ, I. ; KEIDEL, K. ; KLAGES, Albrecht ; KOERBER, Bernhard ; MEYER, M. ; PESCHKE, Rudolf ; PFLÜGER, Jörg ; REINEKE, Vera ; SCHUBERT, Sigrid: Veränderte Sichtweisen für den Informatikunterricht GI-Empfehlungen für das Fach Informatik in der Sekundarstufe II allgemeinbildender Schulen. In: TROITZSCH, Klaus G. (Hrsg.): *Informatik als Schlüssel zur Qualifikation*. Berlin, Heidelberg : Springer, 1993 (Informatik aktuell). – Gesellschaft für Informatik e. V., S. 205–218
- [Schwill 1993] SCHWILL, Andreas: Fundamentale Ideen der Informatik. In: *Zentralblatt für Didaktik der Mathematik* 25 (1993), Nr. 1, S. 20–31. – [http://ddi.cs.uni-dortmund.de/lehre/hauptstudium/ws2001/fundamentale\\_ideen.pdf](http://ddi.cs.uni-dortmund.de/lehre/hauptstudium/ws2001/fundamentale_ideen.pdf) und <http://www.informatikdidaktik.de/Forschung/Schriften/ZDM.pdf>
- [Schwill 1999] SCHWILL, Andreas (Hrsg.): *Informatik und Schule - Fachspezifische und fachübergreifende didaktische Konzepte*. Berlin : Springer, September 1999 (Informatik aktuell)
- [Shaw 1992] SHAW, Mary: *We Can Teach Software Better*. <http://spoke.compose.cs.cmu.edu/shaweb/edparts/crn.htm>. September 1992. – Computing Research News, 4, 4, September 1992, pp. 2, 3, 4, 12; Reprinted in Journal of Computer Science Education, 7, 3, Spring 1993, pp. 4-7
- [van Weert u. a. 1994] VAN WEERT, Tom ; BOSLER, Ulrich ; GUBO, Sam ; TAYLOR, Harriet ; ABAS, Zoraini W. ; DUCHÂTEAU, Charles ; MOREL, Raymond ; WAKER, Peter ; IFIP (Hrsg.) ; UNESCO (Hrsg.): *Informatics for secondary education: a curriculum for schools*. Paris : UNESCO, 1994. – <http://unesdoc.unesco.org/images/0009/000973/097323e.pdf> : Produced by working party of the IFIP under auspices of UNESCO. Paris
- [von Weizsäcker 1971] VON WEIZSÄCKER, Carl F.: *Die Einheit der Natur*. München : Carl Hanser Verlag, 1971
- [Wirth 1999] WIRTH, Niklaus: An Essay on Programming / Eidgenössische Technische Hochschule, Institut für Computersysteme. Zürich, March 1999 ( 315 ). – Forschungsbericht
- [Witten und Penon 1997] WITTEN, Helmut ; PENON, Johann: Unterrichtlicher Einsatz der Telekommunikation. Erfahrungen und Perspektiven aus der Sicht des Informatikunterrichts. In: **[Hoppe und Luther 1997]**, S. 164–175