

Informatik Lernlabor - Systemorientierte Didaktik in der Praxis

Johannes Magenheim

Institut für Informatik
Fakultät für Elektrotechnik, Informatik und Mathematik
Universität Paderborn
Fürstenallee 11
D-33102 Paderborn
jsm@uni-paderborn.de

Abstract: Ausgehend von wichtigen Grundsätzen einer systemorientierten Didaktik der Informatik beschreibt der Beitrag die Konzeption des Informatik Lernlabors. Es wird gezeigt, dass sich die von der systemorientierten Didaktik für die Informatik begründete Methode der Dekonstruktion in besonderem Maße mittels interaktiver computerbasierter Medien realisieren lässt. Das Informatik Lernlabor wird dabei als virtuelle Lernumgebung charakterisiert, in dem informatikbezogene Lernprozesse in Form von ‚Blended Learning‘ organisiert werden. Aus der Perspektive der Didaktik der Informatik ist das Informatik Lernlabor zugleich Gegenstand empirischer Lehr-Lernforschung.

1 Einleitung

Die zunehmende Nutzung von computerbasierten Medien in Lehr- Lernprozessen verschiedenster Fachgebiete konfrontiert auch die Didaktik der Informatik mit der Frage, welche Bedeutung den Informations- und Kommunikationstechniken nicht nur als Unterrichtsgegenstand sondern auch als Medium in der informatischen Bildung beigemessen wird. Hier hat der Einsatz des Computers, z. B. im Informatikunterricht bei der Codierung von Programmen, im Vergleich zu anderen Fächern eine lange Tradition. Mit den erweiterten medialen Funktionen des Computers veränderte sich auch die Palette der Einsatzszenarios von IuK-Techniken in der informatischen Bildung. Diese im Vergleich zum traditionellen Einsatz von Computern erweiterte mediale Funktion kommt besonders jenen didaktischen Ansätzen und methodischen Vorgehensweisen entgegen, die die Visualisierung von Prozessen und Zusammenhängen im Kontext von Informatiksystemen zu deren Verständnis und ihrer interaktiven Modellierung im Unterricht nutzen wollen. Die erweiterten medialen Funktionen des Computers im Informatikunterricht ermöglichen zugleich auch eine größere Bandbreite methodischer Variationen und eröffnen vielfältige Sichtweisen auf die Genese und Praxis sozio-technischer Informatiksysteme.

Die folgende Abhandlung versucht, dieses Postulat zu begründen. Ausgehend von der Perspektive einer systemorientierten Didaktik der Informatik auf sozio-technische Informatiksysteme wird der Einsatz von computerbasierten Medien insbesondere im Informatikunterricht beschrieben und die daraus resultierenden methodischen Implikationen erörtert.

Einer knappen Darstellung von Gegenstandsbereichen, zentralen Zielsetzungen, typischen Unterrichtsmethoden und Anforderungen an digitale Medien folgt im anschließenden Kapitel eine Beschreibung der Konzeption des Informatik Lernlabors mit seinen wesentlichen medialen und methodischen Elementen. Das Informatik Lernlabor wird dabei als eine methodische Variante bei Vermittlung informatischer Lerninhalte beschrieben, die in besonderem Maße die Funktionen einer interaktiven, multimedialen Lernumgebung nutzt.

2 Essentials einer systemorientierten Didaktik der Informatik

2.1 Gegenstandsbereiche – Kriterien zur Inhaltsauswahl

Die systemorientierte Didaktik geht von der Annahme aus, dass es eine der zentralen Aufgaben der Informatik ist, Informatiksysteme (IS) zu gestalten. Infolge dessen werden im Sinne des didaktischen Prinzips der Wissenschaftsorientierung zentrale Gegenstandsbereiche informatischer Bildung diesen Gestaltungsprozessen zugeordnet bzw. von diesen abgeleitet. Unter einem sozio-technischen Informatiksystem (IS) wird dabei die Einheit von Software, Hardware und assoziiertem sozialen Handlungssystem von Personen verstanden, die mit dem technischen Teil des Systems und miteinander interagieren. Zur Software zählt hierbei insbesondere auch die grafische Benutzungsoberfläche (GUI – graphical user interface) während der Hardware auch elektronische und mechanische Bauteile zur Steuerung peripherer technischer Prozesse (embedded systems) und zur Kommunikation mit anderen Informatiksystemen (Vernetzung) zugeordnet werden können. Es ist zu beachten, dass eine eindeutige Abgrenzung der Komponenten eines IS nicht immer möglich ist, da z.B. Teile einer Software in einem Produkt auch in Form von Hardware realisiert werden könnten. Der technische Teil des Systems ist unauflöslich mit dem sozialen Handlungssystem verbunden. Moderne Verkehrsmittel und Verkehrsleitsysteme können ebenso als IS angesehen werden, wie Fertigungsstrassen, Warenwirtschaftssysteme oder eine Lernplattform. Die Software eines IS repräsentiert fundamentale Ideen und fachwissenschaftliche Methoden der Informatik und enthält Konzepte von Arbeitsabläufen, die den sozialen Kontext des Einsatzumfeldes des IS mit den dort angelegten Handlungsrollen von Personen beeinflussen.

Zur Förderung des Verständnisses und der Fähigkeit zur Gestaltung von Informatiksystemen sollte informatische Bildung Kenntnisse über wesentliche informatische Konzepte und Methoden der Systemgestaltung vermitteln. Die

Gegenstandsbereiche, aus denen sich Inhalte für informatische Bildungsprozesse erschließen, lassen sich aus der historischen Entwicklung des Faches mit seinen Wurzeln in der Mathematik und den Ingenieurwissenschaften herleiten. Die Softwaretechnik als zentrale Disziplin der Systemgestaltung, die sich aus diesen historischen Bezügen entwickelt hat und Teilbereiche der Kognitions- und Medienwissenschaften, die Interaktionen des Menschen untereinander und mit der Maschine zum Gegenstand haben, arrondieren das wissenschaftliche Bezugssystem für informatische Bildung. Nicht alle der in Tab 1 benannten Gegenstandsbereiche eignen sich in gleichem Maße als Inhalte informatischer Bildung in unterschiedlichen Bildungseinrichtungen. Speziell für den Informatikunterricht an allgemein bildenden Schulen sind bei der Auswahl von Inhalten weitere Kriterien anzuwenden. Hierzu zählen etwa Prinzipien wie, Wissenschaftsorientierung (relevante Gegenstandsbereiche und Methoden der Fachwissenschaft im Sinne der Wissenschaftspropädeutik), Schülerorientierung (Altersgemäßheit, Orientierung an Schülerinteressen), Erfahrungsorientierung (Gegenstandsbereich ist persönlicher Erfahrung der Schüler zugänglich) oder Handlungsorientierung (Gegenstände lassen sich in einem hohem Maße durch unterrichtliche Eigenaktivität der Schüler erschließen) [Nr99], [Ha03]. Wichtige Anhaltspunkte zur Gewinnung von Inhalten liefert auch das Konzept der fundamentalen Ideen [Sc03].

(A) Mathematik / Theor. Informatik	(B) Ingenieurwissenschaften	(C) Softwaretechnik	(D) Kognition und digitale Medien
<ul style="list-style-type: none"> - Formale Logik - Algorithmen - Formale Beschreibungen u. Kalküle - Numerik - Zahlentheorie - Sprache, Grammatik, Automat - Berechenbarkeit; Komplexität - Turing Maschine - 	<ul style="list-style-type: none"> - formale Typografie - Maschinisierung von Kalkülen - Techn. Semiotik - Protokoll - embedded systems - Mechatronik - Hardware, Rechnerarchitektur - Vernetzung - 	<ul style="list-style-type: none"> - Vorgehensmodelle - Projektmanagement - Modellierungstechniken - Entwurf / Design - HCI / GUI - graf. Beschreibungsmittel - Re-engineering - Systemgestaltung - Programmiersprache - 	<ul style="list-style-type: none"> - Daten - Information - Wissen - Kommunikation - Lernen, Arbeiten - Kooperation - Organisation - Medien - Lernplattformen - Wissensmanagement -

Abb. 1 Gegenstandsbereiche informatischer Bildung in Orientierung an der Genese der Fachwissenschaft und an für die Systemgestaltung relevanten Teilgebieten

2.2 Zielsetzungen – Beitrag zur Allgemeinbildung

In der Didaktik der Informatik kommt neben der Auswahl von relevanten Fachinhalten und den Vermittlungsmethoden insbesondere beim Informatikunterricht an allgemein bildenden Schulen der Frage nach dem Beitrag des Faches zur Allgemeinbildung eine

besondere Bedeutung zu. Allgemeinbildung kann in der Tradition bildungstheoretischer Didaktik in dreierlei Hinsicht charakterisiert werden: allgemein, im Sinne einer Bildung für alle Mitglieder einer Gesellschaft; allgemein, in dem alle Grunddimensionen menschlicher Fähigkeiten und Fertigkeiten durch Lernprozesse angesprochen werden; schließlich allgemein, im Sinne der erkenntnismäßigen Erschließung epochaltypischer Schlüsselprobleme einer Gesellschaft, wie etwa das der Entwicklung und des Einsatzes von IuK-Technologien mit ihren sozialen Folgewirkungen [K196].

Ausgehend von den oben beschriebenen Gegenstandsbereichen informatischer Bildung, deren Aufzählung in Abb. 1 keineswegs vollständig ist und die inhaltlich fachwissenschaftlichen Entwicklungen jeweils angepasst werden muss, können allgemein bildende Funktionen des Informatikunterrichts zur Erreichung von Bildungsstandards verstanden werden. Standards werden hierbei als Kompetenzen und Qualifikationsziele für Schülerinnen und Schüler angesehen, die man mittels Verknüpfung von Inhalten (zentrale Wissensbereiche) und Fähigkeiten (zentrale Kompetenzbereiche) charakterisieren kann. [Te00]

Die in Abb. 2 benannten Kompetenzen zu deren Erwerb u. a. der Informatikunterricht einen Beitrag zu leisten vermag, können zwar schwerpunktmäßig bestimmten Inhaltsbereichen zugeordnet werden. Da diese Zuordnung aber nicht disjunkt und eindeutig ist und der Kompetenzerwerb in verschiedenen Inhaltsbereichen stattfinden kann, soll hier lediglich eine nicht vollständige Benennung erfolgen.

Inhaltsbereich A	B	C	D
<ul style="list-style-type: none"> - Fähigkeit zu vernetztem problemlösendem Denken - Fähigkeit zum Anwenden von formalisierenden Methoden zur Strukturierung von Problemen und zur Modellbildung - Kenntnis der Prinzipien und Unterschiede zwischen maschineller Datenverarbeitung und menschlicher Informationsverarbeitung - Einsicht in Methoden, Beherrschbarkeit und Grenzen der Automatisierung geistiger Prozesse - Kenntnis von Kriterien und Verfahren zur sozialverträglichen Technikgestaltung - Kenntnis über die Funktion technischer Artefakte als externes Gedächtnis und Medium - Fähigkeit zur Nutzung von IuK-Systemen zur Arbeits- und Lernorganisation, zur Kommunikation und zum Wissensmanagement - Kenntnis der gesellschaftlichen Bedeutung und Wirkung von computerbasierten Medien 			

Abb. 2. Möglicher Beitrag von Lernprozessen in der informatischen Bildung zur Allgemeinbildung

2.3 Methodik – Konstruktion und Dekonstruktion

Methodik des Informatikunterrichts orientiert sich in Lerneinheiten, die die Erstellung von Teilen eines Computerprogramms oder eines kompletten Softwareprodukts zum Ziel haben, oft an grundlegenden Vorgehensmodellen der Softwareentwicklung. Einer Phase der Problemanalyse und der Problemeingrenzung folgen Anforderungsdefinition, Designentwurf, Codierung, Implementation und Praxistest des Produkts. Diese von

einigen didaktischen Ansätzen favorisierte unterrichtsmethodische Vorgehensweise der Konstruktion von Software kann für den Informatikunterricht als grundsätzlich geeignet angesehen werden, da sie eine Reihe von lerntheoretischen und lernpsychologischen Aspekten berücksichtigt. Dazu gehören Konzepte wie das der Handlungsorientierung im Unterricht oder die Förderung der Motivation der Schülerinnen und Schüler, die durch das Erlebnis der Materialisierung einer Idee – von dem theoretischen Entwurf hin zum fertigen Softwareprodukt – exemplarisch Kenntnisse über den Prozess der Gestaltung von Informatiksystemen erlangen können. Allerdings beinhaltet dieses unterrichtsmethodische Vorgehensmodell auch eine Reihe von Mängeln:

- die Komplexität der in einer konstruktiven Phase von den Schülern zu erstellenden Software ist oft nicht hinreichend, um informatische Konzepte, wie etwa das der Objektorientierung, hinreichend verdeutlichen zu können.
- der Phase der Modellierung, deren Wichtigkeit immer wieder betont wird [Hu00] wird nicht zuletzt mangels geeigneter medialer Unterstützung und methodischer Konzepte zu wenig Aufmerksamkeit gewidmet.
- Konstruktion von Software im Unterricht erlaubt nur einen eingeschränkten Blick auf die verschiedenen Phasen des Software life cycle. Insbesondere die Revision von Entwurf und Design nach Phasen der Rückkopplung mit Anwendern und Auftraggebern, wie sie bei iterativen partizipativen Vorgehensmodellen üblich sind, können bei einer am Wasserfallmodell orientierten unterrichtlichen Vorgehensweise nur eingeschränkt oder gar nicht berücksichtigt werden.
- Phasen der praxisbezogenen Anforderungsdefinition mit Kopplung an die Anforderungsszenarios von realen Informatiksystemen mit analoger Funktionalität sind nur schwer in das rein konstruktive unterrichtliche Vorgehensmodell integrierbar.
- Phasen der Evaluation der Software in der Praxis und damit die Rückkopplung von Softwareentwicklung mit Situationen des realen Lebens sind in dieses Unterrichtskonzept nur schwer einzubinden.

Um den geschilderten Mängeln eines rein konstruktiv ausgerichteten unterrichtsmethodischen Vorgehens entgegenzuwirken und im Informatikunterricht verschiedene Stadien des Softwareentwicklungsprozesses erfahrbar zu machen, incl. eines zumindest medialen Bezuges zu realen Informatiksystemen, bedarf es medialer und unterrichtsmethodischer Ergänzungen der Konstruktion.

Phasen eines durch multimediale Elemente angereicherten erkundenden und entdeckenden Lernens eines bereits bestehenden und ausführlich dokumentierten Systems, können sich mit Phasen der Konstruktion eines eigenen Softwareprodukts abwechseln. Dabei muss keineswegs die vorliegende ‚fertige‘ Software komplett analysiert und erkundet werden. Es genügt ggf. auch, geeignete informatische Konzepte, wie Algorithmen, Designmodelle oder Codekonstrukte zu analysieren, um sie dann auf Anwendungssituationen bei der Konstruktion des eigenen Softwareprodukts zu übertragen.

Damit ist das unterrichtsmethodische Vorgehen entsprechend den Bedürfnissen der Lerngruppe und den didaktischen Erfordernissen skalierbar. Derart organisierte

Lernprozesse finden ihre Begründung auch in einer Reihe von lerntheoretischen Konzepten [Se00]

- exemplarisches Prinzip und fallbasiertes Lernen: Lernen an Beispielen mit anschließenden Transfer in einen inhaltlich und strukturell verwandten Problemkontext
- Problembasiertes, situiertes Lernen
- Entdeckendes Handlungsorientiertes Lernen unter Integration von multimedialen interaktiven Medienelementen
- Wechsel zwischen konstruktivistischen Formen des Lernens und eher instruktional organisierten Lernszenarien, wobei der Lehrer / die Lehrerin in fortgeschrittenen Phasen der Lerneinheit und in Abhängigkeit von der Vorerfahrung der Lerngruppe eine eher moderierende Rolle nach dem Konzept des cognitive apprenticeship einnehmen kann.

Eine geeignete Alternative zum konstruktiven unterrichtsmethodischen Vorgehen, die diesen Anforderungen genügt, ist das Konzept der Dekonstruktion. Dekonstruktion ist als wissenschaftliche Methode ursprünglich in der Philosophie, der Literaturwissenschaft, später auch in Architektur und Kunst anzutreffen. Die Methode operiert mit spezifischen Formen der Textanalyse und hat die Offenlegung von Form, Inhalt und Hintergründen eines literarischen Werkes und der Intentionen seiner Autoren zum Ziel. Wenn man Software zunächst als Text (Quellcode) betrachtet, mit dessen Hilfe man Maschinen steuern und auf diese Weise implizit auch soziale Wirklichkeit gestalten kann, dann wird klar, dass Dekonstruktion [Zi94] zunächst als Instrument der Quellcodeanalyse auch hier eingesetzt werden kann.

Dekonstruktion ist als Methode der informatischen Bildung in der Lage, vielschichtige Sichten auf Software zu eröffnen und so den Lernenden auf der Basis erster Vorkenntnisse, die sie in traditionellen Vermittlungsprozessen erworben haben, einen differenzierten Einblick in fachwissenschaftliche Konzepte und Methoden zu vermitteln. Bei der Analyse des Quellcodes können nicht nur Klassen, Objekte, Algorithmen oder Sprachkonstrukte einer Programmiersprache erkundet werden, sondern es ergeben sich auch Einblicke in fundamentale informatische Ideen wie z. B. das Konzept ‚Teile und Herrsche‘. Über visualisierte Formen der Softwaredarstellung, wie etwa UML-Diagramme, eröffnen sich Einsichten in komplexe Zusammenhänge und erschließen sich möglicherweise verwendete Standard-Entwurfsmuster, sowie Design- und Entwurfsentscheidungen. Eine Analyse der GUI ermöglicht Zugänge zu Problemen der Softwareergonomie aber auch zu Konzepten wie IO- und Exception-Handling. Die Funktionalität der Software kann insgesamt getestet und bewertet werden. Möglicherweise sind verschiedene Systementwürfe verfügbar, die miteinander verglichen und hinsichtlich ihrer informatischen Konzepte und Folgewirkungen im sozialen Einsatzkontext bewertet werden können. Letzteres setzt aber eine solide informatische Wissensbasis hinsichtlich Inhalten und fachwissenschaftlichen Methoden sowie das Erarbeiten von adäquaten Bewertungskriterien voraus und ist somit als optionales

fortgeschrittenes Lernziel anzusehen. Damit können sich über Dekonstruktion nicht nur Einblicke in das vorliegende Softwareprodukt sondern auch in den Prozess seiner Genese eröffnen. ‚Protokolle‘ über Designentscheidungen, die sich auf die Bewertung alternativer Entwürfe beziehen können verdeutlichen, wie Nutzungsszenarien und Handlungsabläufe im sozio-technischen Informatiksystem festgelegt werden. Die Diskussion von Fragen der kontextuellen Informatik [Ki02] ist somit kein Additum, das gegebenenfalls als Exkurs an eine Lerneinheit angefügt wird, sondern ist im informatischen Kernbereich der Anforderungsanalyse, des Systemsentwurfs und des Softwaredesigns angesiedelt. Ob ein Informatiksystem die Erhebung von Leistungsdaten an Arbeitsplätzen ermöglicht ist einerseits eine Entwurfs- bzw. im Detail eine Designentscheidung und hat andererseits erhebliche Bedeutung für den betrieblichen Controllingprozess und den Datenschutz. Das Verhältnis von Produkt und Prozess bei der Softwareentwicklung wird damit zum relevanten Gegenstand im Informatikunterricht anhand dessen sich elementare Konzepte der Informatik erarbeiten lassen [Ma03].

2.4 Medien – ‚Cognitive Tools‘, didaktische Software und ‚Learning Objects‘

Das Informatik Lernlabor stellt den Versuch dar, dieses unterrichtsmethodische Konzept unter wissenschaftlicher Begleitung in die Praxis umzusetzen. Dazu bedarf es einer Reihe von medialen Voraussetzungen. Neben den in der informatischen Bildung schon seit jeher verwendeten Tools zur Softwareentwicklung wie z.B. Editoren, Debugger, oder Compiler sollten in dekonstruktiven Unterrichtsphasen eine Reihe weiterer interaktiver computerbasierter Medien eingesetzt werden. Man kann dabei in Anlehnung an Keil-Slawiks Definition von primären, sekundären und tertiären Medienfunktionen [Ke02] zwischen Medien verschiedenen Typs unterscheiden:

- ‚Cognitive Tools‘ ermöglichen die interaktive Gestaltung und Strukturierung von Dokumenten, Software und computerbasierten Medien. Durch geeignete Repräsentationen und Anordnungen gewährleisten sie deren Bearbeitung, Übertragung und Sicherung.
- ‚Lernsoftware‘ beinhaltet neben der reinen Inhaltspräsentation eine didaktisch und lerntheoretisch motivierte Abfolge von Interaktionen und Rückmeldungen mit den Nutzern. Damit werden Formen der Mediennutzung im Medium selbst abgebildet.
- ‚Agents‘ repräsentieren schließlich einen Medientyp, der Lerner- und Nutzungsverhalten anhand der stattfindenden Interaktionen analysieren und im Hinblick auf Lerneffizienz für die Nutzer modellieren kann. ‚Agents‘ besitzen somit eine gewisse algorithmen- oder regelbasierte ‚Lernfähigkeit‘ und werden deshalb oft auch als ‚intelligente tutorielle Systeme‘ bezeichnet.

Während der Medientyp ‚agent‘ in der bisherigen pädagogischen Praxis eine kaum wahrnehmbare Rolle spielt, sind ‚cognitive tools‘ und ‚Lernsoftware‘ Medientypen, die auch im Informatik Lernlabor zum Einsatz kommen. Bei den Cognitive Tools sind

besonders die grafischen Debugger und die integrierten Entwicklungsumgebungen zu nennen, die die Konstruktion der Software erleichtern. Grafische Editoren, die z.B. das interaktive Erstellen von UML Diagrammen und deren automatische Umwandlung in Quellcode ermöglichen, sind hier ebenfalls zu nennen. Mit ihrer Hilfe könne Phasen der Systemmodellierung interaktiv am Rechner gestaltet, der gesamten Lerngruppe vorgestellt und nach Diskussion auch leicht revidiert werden. Damit stehen im Informatikunterricht gerade für diese wichtige Lernphase im Gegensatz zu früher geeignete mediale Hilfsmittel zu Verfügung.

Als besondere Form der ‚Lernsoftware‘ werden im ILL (Informatik Lernlabor) Lernobjekte (‚learning objects‘) genutzt [Ie02]. Darunter sind abgeschlossene kleinere multimediale Lerneinheiten zu verstehen, die den Nutzern Inhalte mit interaktiven multimedialen Medien präsentieren und die sich auf eine begrenzte Anzahl von zu realisierenden Lernzielen beziehen. Lernobjekte dienen den Nutzern vorwiegend zum selbstgesteuerten erkundenen Lernen eines Gegenstandsbereichs. Ihre Integration in das Informatik Lernlabor ermöglicht ein Konzept von ‚blended Learning‘, bei dem sich Formen des Präsenzlernens mit Phasen des e-learning mit computerbasierten Medien abwechseln [Ma03] [Sa03]. Lernobjekte sollten den Lernenden in einer interaktiven für kollaborative Arbeitsweisen offenen und möglichst webbasierten Erkundungsumgebung angeboten werden. Diese Aufgabe kann z.B. von einer Lernplattform oder Groupware mit spezifischen, das e-learning unterstützenden Funktionen übernommen werden. Eine solche Lernumgebung, die exploratives Lernen unterstützt, wird neben den Lernobjekten weitere multimediale Dokumente und ein zu dekonstruierendes Softwareprodukt enthalten. Zu den ergänzenden Medienobjekten zählen etwa Videosequenzen von Arbeitsabläufen in realen Informatiksystemen, Interviews mit Auftraggebern und Nutzern, Gespräche zwischen Entwicklern über Entwurfsentscheidungen sowie Animationen zu Informationsflüssen und Arbeitsabläufen (use cases). Damit werden Vermittlungskonzepte, wie die videogestützte Anforderungsdefinition umsetzbar. Die zu dekonstruierende Software ist ein in seiner Komplexität reduziertes Produkt, das aber wesentliche Features eines realen Informatiksystems beinhaltet und die Exploration von fundamentalen informatischen Konzepten erlaubt. Tools und Lernsoftware in Form von learning objects ermöglichen vielfältige Sichten auf das Produkt Software und seinen Entstehungsprozess. Durch diese Form der medialen Repräsentation können Anwendungsorientierung und Realitätsbezug der informatischen Bildung gesichert und bei den Studierenden der Erwerb von vernetztem Wissen z.B. auf dem Gebiet der Softwaretechnik gefördert werden. Bleibt die Frage nach dem Lerndesign für den Einsatz digitaler Medien in der informatischen Bildung. Zwar ist Dekonstruktion ohne multimediagestütztes kooperatives Lernen kaum denkbar. Dennoch ist die Frage zu beantworten, wie die multimedialen Lernobjekte in ein sinnvolles Lernszenario eingebettet werden können.

3 Das Informatik Lernlabor

Die bisher weitgehend theoretisch begründete und aus den Erfordernissen einer systemorientierten Didaktik hergeleitete methodische Konzeption des Informatik Lernlabors soll nun in den folgenden Abschnitten konkretisiert werden. Dabei wird in der Darstellung im Wesentlichen auf eines der gegenwärtig in der Entwicklung und Evaluation befindlichen Inhaltsmodule, das ‚Hochregallager‘ zurückgegriffen.

3.1 Blended Learning im Informatik Lernlabor

Die Produktion, der Einsatz und das nachhaltige Management digitaler Medien zu Lehr- und Lernzwecken erfordern konzeptionelle Entscheidungen in verschiedenen Bereichen, die man zusammenfassend als Lerndesign bezeichnen kann. Dies gilt auch für das Lernen in einer vernetzten, interaktiven computerbasierten Lernumgebungen wie dem Informatik Lernlabor. Lernszenarien werden von methodisch-didaktischen und technisch-organisatorischen Komponenten bestimmt. Im didaktischen Kontext einer Lerneinheit oder Lehrveranstaltung sind Entscheidungen über Ziele, Inhalte und methodisches Vorgehen zu treffen. Ebenso sind lerntheoretische Erwägungen, die die Rolle der Lehrenden und Lernenden im Lernszenario betreffen sowie Festlegungen hinsichtlich der erforderlichen Vorkenntnisse der Lernenden für ein erfolgreiches Lernarrangement von zentraler Bedeutung.

Abb. 3 zeigt das Lerndesign im Informatik Lernlabor. Die Lernenden erwerben ihre Kenntnisse über den Lerngegenstand sowohl aus geeigneten Lernobjekten als auch in traditionellen Formen der Präsenzlehre. Die präsentierten Inhalte, insbesondere die der Lernobjekte, beziehen sich sowohl auf informatische Konzepte mit denen das Softwareprodukt realisiert wurde als auch auf den Ablauf von technischen Prozessen, den workflow sowie auf Interaktionen von Mitarbeitern im und mit dem Informatikssystem. Die Lernobjekte, die cognitive tools und weitere inhaltsbezogene Dokumente und Materialien werden den Lernenden in einer multimedialen Erkundungsumgebung mit Groupwarefunktionen bereitgestellt. Sie übernimmt damit wesentliche Aufgaben, die an eine Lernplattform gestellt werden. [Sc03] Ferner muss die Lernplattform die zu dekonstruierende Software sowie geeignete Tools bereitstellen, die verschiedene Views auf die Software ermöglicht (Quellcode, UML-Diagramme, GUI etc.). Beim Beispiel ‚Hochregallager‘ kommt noch das Mindstorm Modell mit den Legokonstruktionen als Quelle von Erkundungen hinzu. Damit sind zunächst grundlegende Voraussetzungen geschaffen, um den Lernenden im Wechsel von entdeckendem, selbstgesteuertem Lernen mit Materialien der Lernplattform und Phasen des Präsenzlernens die in der Lerneinheit angestrebten informatikbezogenen Kompetenzen zu vermitteln.

Lerndesign im Informatik Lernlabor (ILL)

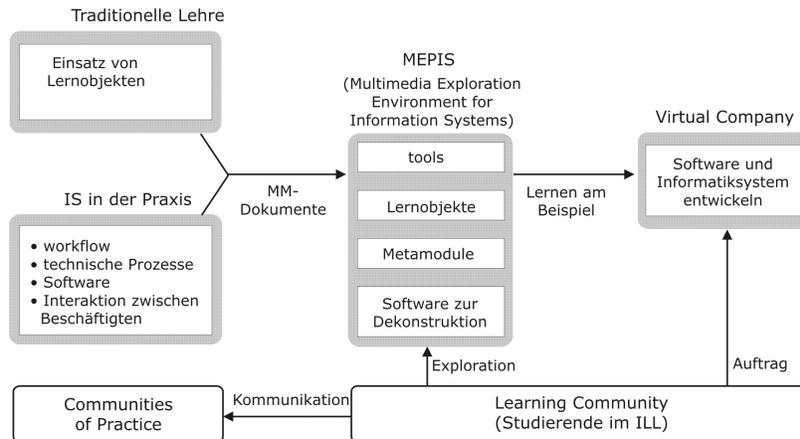


Abb. 3 Das Lerndesign im Informatik Lernlabor

Um erfolgreiches Blended Learning in diesem Lerndesign zu garantieren bedarf es jedoch zusätzlicher methodischer Konzepte, die sich aus lerntheoretischen Überlegungen zum blended learning herleiten. [Hu02] So kann beispielsweise ein umfassender Gestaltungsauftrag, wie wir ihn im Informatikunterricht aus dem projektorientierten Lernen kennen, zur Bildung von Lerngruppen führen, die im Rahmen dieses Auftrages zunehmend selbstständig ihre Arbeit verrichten [Vw01]. Beispielsweise erhalten die Lerngruppen im Szenario ‚Hochregallager‘ den Auftrag zur Gründung einer virtuellen Firma, die nach Studium der Technik des Lagers eine automatisierte Packstrasse in der Lego Mindstorm Technik entwickeln soll. Die Lernenden bilden im Idealfall eine ‚learning community‘, die ihre Wissensbeschaffungs- und Konstruktionsprozesse weitgehend selbstständig organisiert und ggf. auch zusätzliche Impulse für ihren Auftrag durch Erkundung eines entsprechenden realen Informatiksystems erhalten könnte.

Lernprozesse im Informatik Lernlabor können nach diesem unterrichtsmethodischen Vorgehensmodell in unterschiedliche Phasen eingeteilt werden, die in Abhängigkeit von den konkreten Gegenstandsbereichen variieren. In Bezug auf das Beispiel ‚Hochregallager‘ ergeben sich die in Abb 4 dargestellten Phasen. Diese Arbeitsphasen ähneln in manchen Abschnitten denen, des traditionellen Projektunterrichts. Dort fehlen allerdings die Dekonstruktions- und Transferphasen und das Konzept des ‚blended learning‘. Aus diesem Vergleich ergeben sich auch weitere methodische Hinweise. Lernen im Informatik Lernlabor in dem hier vorgestellten Umfang eignet sich nicht für den Anfangsunterricht, es sei denn, man möchte gezielt einzelne informatische Konzepte erkunden und nutzt entsprechende Lernobjekte oder Views auf die vorliegende Software.

- Produktionsauftrag' an die virtuelle Firma mit einer detaillierten Anforderungsbeschreibung des zu konstruierenden Systems. In unserem Beispiel wäre dies eine Packstrasse, die in der Lage ist, Paletten in Abhängigkeit von den jeweiligen Bestellungen mit unterschiedlichen Produkten aus einem Gesamtsortiment zu bestücken.
- Erste Erkundungen der Funktionalität des Systems. Dieser Prozess kann durch geeignete Erkundungsaufträge mit Hinweis auf adäquate Materialien erheblich beschleunigt werden.
- Modellieren der Software des Systems mit CRC – Karten seitens der Lernenden. Transfer der Ergebnisse in UML-Klassendiagramme [Am98]
- Erkunden der vorgegebenen Modellierung und der Art ihrer Implementierung in der zu dekonstruierenden Software anhand von geeigneten UML-Diagrammen (Klassen, Attribute, Methoden Objekte Attributwerte). Vergleich mit den selbst entworfenen Konzeptionen.
- Erwerben von vertieften Kenntnissen über die zu dekonstruierende Software und deren Wirkung auf das Informatiksystem auf konzeptioneller und auf Sourcecodeebene. Hierbei sollen auch Erkenntnisse über grundlegende informatische Konzepte gewonnen werden. Gezielte Erkundungsaufträge sind in dieser Phase unerlässlich.
- Ein begrenzter Re-engineeringsauftrag zur Veränderung der vorliegenden Software und der Funktionalität des Informatiksystems hilft die notwendigen Kenntnisse und Fertigkeiten zu erweitern, die später zur Konstruktion des eigenen Softwareprodukts benötigt werden. Dabei werden Eingriffe auf UML- und Sourcecodeebene notwendig. Beim Beispiel des Hochregallagers sind auch Veränderungen in der Mechatronik des embeded systems erforderlich (Änderung bei den eingesetzten Sensoren und der erforderlichen Flexibilität des Regalbediengeräts).
- Beginn der Konstruktionsphase für das Softwareprodukt der virtuellen Firma. Mittels Transferprozessen können exemplarische Lösungsansätze, wie sie in der dekonstruierten Software entdeckt wurden, auf das neue Aufgabenszenario übertragen werden. Der Transfer erfolgt sowohl auf der Design- und Quellcodeebene als auch- in unserem Beispiel auf mechatronischer Ebene (Sensorik, Mechanik)
- Modellierung des zu entwickelnden Informatiksystems. In unserem Beispiel sowohl auf technischer Ebene bei der mechanischen Konstruktion (z.B. Konzepte der Kraftübertragungen) als auch auf Softwareebene bei der Prozesssteuerung (Zusammenspiel von Sensoren und Aktoren, Kommunikation zwischen den einzelnen mindstorm Bausteine)
- Arbeitsteilige Implementierung des kompletten Systems und Testen von dessen Funktionalität, ggf. Optimierung des Informatiksystems.
- Zum Abschluss werden sowohl die Qualität des geschaffenen Produkts als auch der Lernprozess im Informatik Lernlabor incl. der entstandenen Schwierigkeiten reflektiert

Abb. 4 Arbeitsphasen im Informatik Lernlabor

Effizientere Lernprozesse sind zu erwarten, wenn die Mitglieder der Lerngruppe bereits über fundamentale Vorkenntnisse in der Modellierung und in der Generierung von Sourcecode verfügen. Das Konzept des Informatik Lernlabors wird gegenwärtig mit entsprechendem gesteigertem Anspruchsniveau auch in der Hochschullehre zur Softwaretechnik eingesetzt und evaluiert.

3.2 Mediale Elemente des Informatik Lernlabors

Wesentliche mediale Elemente des Informatiklernlabors, die über eine multimediale Erkundungsplattform den Lernenden zugänglich gemacht werden, wurden bereits in

Abschnitt 2.4 erwähnt. An dieser Stelle erfolgt die Konkretisierung für das Modul ‚Hochregallager‘.

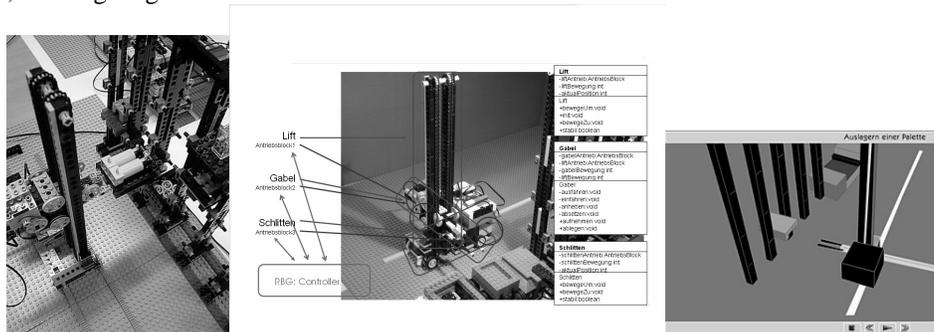


Abb. 5 Mindstorm Modell, Animationen und Videosequenzen zur Erkundung des Hochregallagers

Abb. 5 zeigt einige Elemente der Erkundungsumgebung, die den Lernenden zunächst einen Einblick in die Modellierung eines ‚embedded systemes‘ mit seinen Hard- Firm- und Softwarekomponenten ermöglichen soll. Videos eines realen Hochregallagers mit seinen technischen Funktionen und den Interviews mit den dort Beschäftigten über ihren Arbeitsplatz liefern ergänzende Informationen. Darüber hinaus sind fachbezogene Dokumente über die in dem System enthaltenen informatischen Konzepte zu finden, wie etwa das in Abb. 6 dargestellte Klassendiagramm.

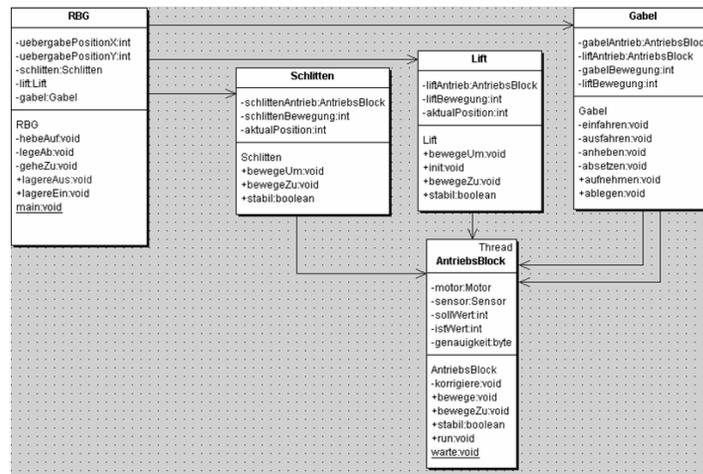


Abb. 6 Klassendiagramm der Steuerungs- und Kommunikationssoftware

Grundsätzlich kann man zwischen folgenden unterschiedlichen Elementen der multimedialen Erkundungsumgebung unterscheiden:

Allgemeine und themenbezogene Dokumente und Lernobjekte (Inhaltsmodule)

Zu den allgemeinen Lernobjekten und Dokumenten gehören Informationen über Protokolle und technische Kommunikation, Algorithmen, Objektorientierte Modellierung, Modellierungssprachen, Entwurfsmuster und Java Sprachkonstrukte. Die spezifischen Lernobjekte und Dokumente beziehen sich auf das der Lerneinheit jeweils zugrunde liegende Inhaltsmodul. Im Fall des Hochregallagers gehören dazu unter anderem Informationen über eingebettete Systeme, die Firmware des Mindstorms RCX, die zugehörige Javaschnittstelle, Prinzipien der Steuerung und Regelung, die Mechanik der Lego Bauteile, von den Mindstorm bricks verwendete Kommunikationsprotokolle sowie Informationen über Elemente eines realen Hochregallagers.

Softwaretools und spezifische Tools

Zu den allgemein genutzten Tools zählen Modellierungswerkzeuge wie UML-Editoren, auf CRC Karten bezogene Mindmapping Tools, grafische Debugger und integrierte Entwicklungsumgebungen. Spezifische Tools sind im Falle des Hochregallagers virtuelle Simulationsumgebungen für die mindstormbricks, sowie mindstormbezogene Entwicklungstools.

Guides tours

Guided tours dienen der Erkundung der multimedialen computerbasierten Lernmaterialien. Sie können von den Lernenden in Phasen des selbstgesteuerten Lernens z.B. anhand von angebotenen Navigationshilfen genutzt werden, um sich Lerninhalte eigenständig zu erschließen, die sie für die Bewältigung der Konstruktionsaufgabe benötigen.

LMS und CMS

Die multimedialen Materialien und Dokumente, die computerbasierten Animationen und digitalen Videosequenzen müssen den Lernenden in geeigneter Form zur Verfügung gestellt werden. Gleiches gilt für die benötigten Cognitive Tools. Dazu wird zumindest eine Groupware mit elementaren Funktionen einer Lernplattform benötigt. Sie sollte über ein User- und Accessmanagement verfügen und elementare webbasierte Kommunikationsdienste anbieten (mail, chat). Auf diese Weise können für das Konzept des blended learning grundlegende Funktionen eines Lernmanagement- (LMS) und Contentmanagementsystems erfüllt werden. In der gegenwärtigen Erprobungsphase des Lernlabors verwendet die Arbeitsgruppe das System Steam [St03].

Didaktische Software

Die didaktische Software dient der oben bereits ausführlicher beschriebenen Methode der Dekonstruktion. Im Fall des Hochregallagers wird eine Steuerungssoftware für die Ein- und Auslagerung von Paletten mit der dazu notwendigen Auftragsverwaltung, den Transporten zwischen verschiedenen Bereichen des Lagers und der Kommunikationskontrolle zwischen den autonomen Systemen bereitgestellt. Außerdem gehört ein mit Legobausteinen konstruiertes Modell eines Hochregallagers, das von der Software gesteuert wird, zu dem zu erkundenden didaktisch aufbereiteten Informatiksystem.

Aufgabensammlung und Erkundungsaufträge

Die Erkundung des Hochregallagers wird jeweils durch spezifische Erkundungsaufträge initiiert, die sich auf wesentliche informatische Prinzipien des Systems beziehen und für die später zu lösende komplexere Konstruktionsaufgabe von zentraler Bedeutung sind. Hierzu zählt beispielsweise die Analyse und Bewertung des von den autonomen Systemen verwendeten Kommunikationsprotokolls, das mit gängigen Konzepten wie Token Ring und CSMA/CD im Hinblick auf seine Effizienz verglichen werden kann. Kleinere themenbezogene Übungsaufgaben in der Erkundungsumgebung sind darüber hinaus geeignet, während der Dekonstruktion gewonnene Einsichten und Fertigkeiten zu vertiefen.

3.3 Inhaltsmodule des Informatik Lernlabors

Das Informatik Lernlabor verfügt gegenwärtig über zwei Inhaltsmodule, deren Grad des Ausbaus einen ersten Einsatz in Unterricht und Hochschullehre zulassen. Ein weiterer befindet sich im Aufbau, ein vierter schließlich in der Planung. An dieser Stelle kann aus Platzgründen nur ein kurzer Hinweis auf die jeweiligen Inhalte erfolgen.

Modul Hochregallager

- hat die Steuerung von Transport- und Lagerungsprozessen in einem Hochregallager zum Gegenstand. Neben den üblichen computerbasierten Medien wird in diesem Modul zusätzlich ein Lego Mindstorm Modell eines Hochregallagers für Zwecke der Dekonstruktion bereitgestellt. Als Transferaufgabe wird die Konstruktion und Steuerung einer Packstrasse mit Lego Mindstorms Bauteilen erwartet.

Modul Schulkiosk

- in diesem Inhaltsmodul werden elementarste Konzepte eines Warenwirtschaftssystems thematisiert. Die zu dekonstruierende Software bezieht sich auf die Ein- und Verkaufsvorgänge eines Schulkiosks. Verkaufshandlungen werden als use cases analysiert, um geeignete Klassenstrukturen für die Modellierung zu gewinnen. Es werden Systemerweiterungen und Re-engineeringsaufgaben als Transferleistung erwartet.

Modul ‚Computerspiel‘

- das Computerspiel ‚Ursuppe‘ bildet den Ausgangspunkt dieses Inhaltsmoduls. Die zugehörige Software ist in der Lage, nach entsprechenden Benutzereingaben die Spielverwaltung zu übernehmen und Spielstände grafisch anzuzeigen. Anhand des Spiels kann u. a. thematisiert werden, wie bei Entwurfs- und Designentscheidungen zwischenmenschliche Interaktionen und Handlungen graduell auf das System übertragen werden können. Zum Zwecke des Transfers wird die Entwicklung einer Software für ein anderes Spiel erwartet.

Modul ‚Webshop‘

- in diesem, bisher nur in Grundzügen vorliegenden Modul, sollen vor allem webbasierte Transaktionen, die Gestaltung von interaktiven Webseiten und die Speicherung von deren Inhalten in einer Datenbank thematisiert werden.

3.4 ILL und informatikbezogener Kompetenzerwerb

Das didaktisch-methodische Konzept des Informatik Lernlabors kann inhaltlich und im Schwierigkeitsgrad variiert werden und eröffnet deshalb sowohl Einsatzmöglichkeiten in der Informatiklehre an der Hochschule als auch im Informatikunterricht in der Sekundarstufe II. Das Konzept ist lerntheoretisch fundiert und bietet eine motivierende Ergänzung und Alternative zu traditionellen Formen des Lernens im Informatikunterricht. Allerdings sind die medialen Anforderungen und der Aufwand zur Materialproduktion nicht unerheblich. Hinzu kommt, dass die Lernmodule gewartet und inhaltlich stets auf einem aktuellen fachwissenschaftlichen und fachdidaktischen Stand gehalten werden müssen.

Mit den informatikbezogenen Lernprozessen im Informatik Lernlabor lassen sich Kompetenzen auf kognitiver, fachmethodischer, sozial-kommunikativer und normativ-bewertender vermitteln. Obwohl eine Zuordnung von Kompetenzen zu einzelnen Ebenen nicht immer eindeutig ist lassen sich exemplarisch folgende Zuweisungen vornehmen:

kognitive Ebene: z. B. Wissenserwerb über Sprach- und Designkonstrukte, Kenntnisse einer Programmiersprache (z.B. Java) und von Modellbeschreibungssprachen (z.B. UML), Kenntnisse über Kommunikationsprotokolle, Prinzipien der Messwerterfassung, und des Steuerns und Regelns technischer Systeme, Einsicht in den Produkt-Prozesscharakter von Software und die Ursachen von Designkonflikten...

fachmethodischer Ebene: z.B. Beherrschung von Vorgehensmodellen in der Softwaretechnik, von Modellierungstechniken, Problemlösetechniken, Methoden des Softwareentwurfs...

sozialkommunikativer Ebene: Umgang mit virtuellen Lernumgebungen, kooperatives Lernen und Arbeiten im Team, Techniken der Ergebnispräsentation...

normativ bewertender Ebene: Bewertung der Qualität von Entwürfen und von Softwaredesign, Beurteilungsfähigkeit hinsichtlich der sozialen Folgewirkungen von Entwurfs- und Designentscheidungen bei der Systemgestaltung...

3.5 Wissensmanagement im Informatik Lernlabor

Um die für das ILL erzeugten Dokumente und Lernobjekte effizient in verschiedenen Lernszenarien einsetzen zu können empfiehlt es sich, sie in ein Konzept von Wissensmanagement einzubinden. Dies ergab sich für das hier vorgestellte Projekt auch aus der Tatsache, dass die Materialien im Rahmen von zwei BMBF-Projekten entwickelt wurden, an denen die Arbeitsgruppe Didaktik der Informatik beteiligt ist und in denen Strategien des Wissensmanagements zum Einsatz kamen¹.

¹ Es handelt sich um die Projekte MuSoft: <http://www.musoft.org>
und SIMBA <http://www.die.et-inf.uni-siegen.de/simba/foerderung.html>

Wissensmanagement umfasst die Repräsentation (Darstellung), die Kommunikation (Verteilung, Verbreitung), die Generierung (Förderung von Lernprozessen) und die Nutzung (Anwendung, Handlung) von Informationen und Wissen [MSN01]. Um die Lernobjekte und Dokumente hinsichtlich ihrer technischen und vor allem didaktischen Anforderungen zu strukturieren, werden sie gegenwärtig mit Metadaten in Anlehnung an den LOM-Standard beschrieben [Ie02]. Darüber hinaus werden wegen der erforderlichen Kompatibilität beim möglichen Datenaustausch zwischen verschiedenen Lernplattformen Vorgaben weiterer Standardisierungsinitiativen hinsichtlich Schnittstellen wie z.B. dem IMS Global Learn Consortium, SCORM oder ARIADNE [Ma03] beachtet.

Außerdem wurde eine Hierarchisierung der Lernobjekte eingeführt, die ihre Zuordnung zu Modulen und ihre Wiederverwendbarkeit erleichtern soll. Lernobjekte werden im Informatik Lernlabor in einer vierstufigen Hierarchie gegliedert. Lerneinheiten stellen die oberste Hierarchieebene dar und bestehen aus verschiedenen Lernmodulen, die ihrerseits wiederum Gruppenobjekte enthalten, welche letztlich aus Medienobjekten (z.B. Animationen, Folien, Grafiken, Videos...), den atomaren Einheiten für Lernobjekte zusammengesetzt sind. Der Begriff Lernobjekt wird dabei als Abstraktion verwendet, der konkrete Objekte unterschiedlicher Hierarchieebenen repräsentieren kann. Bei der Gestaltung von Lerneinheiten, müssen nicht notwendigerweise alle vier Ebenen durchgängig genutzt werden.

3.6 Praxisintegration von didaktischer Forschung

Die im Informatik Lernlabor entwickelten didaktisch methodischen Konzepte müssen im Hinblick auf ihre Tauglichkeit bei der Vermittlung informatischer Bildungsinhalte und insbesondere hinsichtlich der erhofften lernförderlichen Wirkung des Einsatzes von computerbasierten Medien differenziert empirisch überprüft werden. Dazu sind spezifische empirische Konzepte und Untersuchungsinstrumente der Lehr- Lernforschung entwickelt worden, die teilweise in einer ersten Evaluationsphase bereits eingesetzt wurden. Die erste Erprobung des Konzepts fand mit Lehramtsstudierenden der Informatik im Rahmen eines didaktischen Seminars statt. Als Inhaltsmodul kam das ‚Hochregallager‘ mit den Lego Mindstorm Modellen zum Einsatz. Die Studierenden arbeiteten dabei nach dem oben geschilderten didaktisch methodischen Konzept. Bei der empirischen Auswertung der Lernprozesse wurden in einer Abschlussdiskussion gleichzeitig auch mögliche Probleme beim Transfer des Konzepts in den Informatikunterricht der SekII diskutiert.

In einer ersten evaluativen Studie sind zunächst Eingangsbefragung mit einem Fragebogen und darauf aufbauende Gruppendiskussionen durchgeführt worden. Eine Analyse des Umgangs mit den Softwaretools wurde mittels screenrecording realisiert. Dieses Verfahren sollte neben der Prozessbeobachtung Hinweise auf mögliche Schwierigkeiten in den Phasen des Wissenstransfers in den Problemkontext der Konstruktionsaufgabe liefern. Da sich die Begleitforschung am Konzept einer formativen

Evaluation orientiert, konnte das Seminarkonzept unter Berücksichtigung der gewonnenen empirischen Erkenntnisse phasenweise aktuellen Erfordernissen angepasst werden. Am Ende des Seminars, zum Abschluss der Konstruktionsphase, werden Interviews mit den Studierenden und eine prozessbewertende Gruppendiskussion stehen. Diese Phase der Datenerhebung ist zum Zeitpunkt des Redaktionsschlusses für diesen Artikel noch nicht durchgeführt und ausgewertet worden. Dennoch kann man nach dem bisherigen Seminarverlauf eine positive Zwischenbilanz ziehen, da der Prozess des Wissenserwerbs und -transfers zu einem bemerkenswerten Softwareprodukt führte (Packstrasse) und dabei auch wesentliche informatische Konzepte vermittelt werden konnten.

4 Schlussfolgerungen und weitere Perspektiven

Das Konzept des Informatik Lernlabors muss sowohl hinsichtlich seiner hochschuldidaktischen Eignung als auch im Hinblick auf seine Tauglichkeit für den Informatikunterricht in der gymnasialen Oberstufe weiterentwickelt und evaluiert werden. Die bisher vorliegenden Inhaltsmodule sollen weiter ausgebaut und um die in den didaktischen Seminaren entwickelten Produkte ergänzt werden. Gleichzeitig ist es erstrebenswert, Module die bereits einen Mindeststandard an Lernobjekten, ‚Cognitive Tools‘ und Dokumenten aufweisen, auf einer breiteren Basis mit unterschiedlichen Lerngruppe zu erproben.

Acknowledgement

Mein besonderer Dank für das Zustandekommen dieses Beitrags gilt den Mitgliedern der AG Didaktik der Informatik an der Universität Paderborn und in besonderem Maße Herrn Olaf Scheel ohne deren motiviertem Einsatz die Realisierung des Informatik Lernlabors nicht möglich gewesen wäre.

Literaturverzeichnis

- [Am98] Ambler, S.W.: Building Object Applications That Work, New York , SIGS Books, NY 1998.
- [Ha03] Rahmenplan Wahlpflichtfach Informatik, Bildungsplan neunstufiges Gymnasium Sekundarstufe I, hrsg. v. der Behörde für Bildung und Sport, Freie und Hansestadt Hamburg, Hamburg 2003.
- [Hu00] Hubwieser, P., Didaktik der Informatik - Grundlagen, Konzepte, Beispiele, Berlin u. a. 2000.

- [Hu02] Hung, D. Forging links between 'Communities of Practice' and Schools in Journal of E-Learning, April–June 2002 , Vol. 1, No.2, (pp. 23–33).
- [Ie02] IEEE Learning Technology Standards Committee (2002), IEEE, 3 Park Avenue New York, NY 10016-5997, USA. Final Draft of the IEEE Standard for Learning Objects and Metadata, June 2002. Online: <http://ltsc.ieee.org/wg12/>
- [Ke02] Keil-Slawik, R. Denkmedien – Mediendenken: Zum Verhältnis von Technik und Didaktik. In: it + ti. Informationstechnik und Technische Informatik 44. Jg. H.4. (pp 181 – 186): Oldenbourg, 2002.
- [Ki02] Projekt Kontextuelle Informatik: <http://iug.uni-paderborn.de/iug/projekte/kontextinfo>
- [KI96] Klafki, W. Grundzüge eines neuen Allgemeinbildungskonzepts. Im Zentrum: Epochaltypische Schlüsselprobleme. S. 43 – 81. In: Klafki, Wolfgang: Neue Studien zur Bildungstheorie und Didaktik. Weinheim, Beltz. 5. Auflage 1996 .
- [Ma01] Magenheimer, J. Deconstruction of Socio-technical Information Systems with Virtual Exploration Environments as a Method of Teaching Informatics In Montgomerie, C., Viteli, J. (eds): Proceedings of ED-MEDIA 2001, World conference on Educational Multimedia, Hypermedia & Telecommunications, Tampere Finland 25–30.06.2001 (pp1199–1204).
- [Ma03] Magenheimer, J. Wissensmanagement, Dekonstruktion und 'Learning Communities' in der Softwaretechnik – Didaktische Konzepte im BMBF-Projekt MuSoFT in: Rinn, U. / Wedekind, J. (Hrsg.), Didaktik der neuen Medien. Münster 2003
- [MSN01] Mandl, H., Schnurer, K. Nistor, N. Einführung in das Wissensmanagement, Online: <http://wissman.emp.paed.uni-muenchen.de/>
- [Nr99] Lehrplan Informatik, Sekundarstufe II – Gymnasium und Gesamtschule, hrsg. v. Ministerium für Schule und Weiterbildung , Wissenschaft und Forschung des Landes Nordrhein Westfalen Schriftenreihe Schule in NRW Nr. 4725, Düsseldorf 1999.
- [Sa03] Sauter, W., Sauter A.M. Blended Learning, Effiziente Integration von E-Learning und Präsenztraining, Luchterhand 2003.
- [Sc93] Schwill, A. Fundamentale Ideen der Informatik, Zentralblatt für Didaktik der Mathematik 1 (1993) 20-31.
- [Sc03] Schulmeister, R. Lernplattformen für das virtuelle Lernen, Evaluation und Didaktik, München, Wien: Oldenbourg 2003.
- [Se00] Seel, N. M.: Psychologie des Lernens. Lehrbuch für Pädagogen und Psychologen. UTB, 2000.
- [St03] Structuring Information in a Team. <http://steam.upb.de/index.html>

- [Te00] Terhart, E. (Hrsg.): Perspektiven der Lehrerbildung in Deutschland. Abschlussbericht der von der Kultusministerkonferenz eingesetzten Kommission. Weinheim/Basel 2000.
- [Vw01] van Weert, T.J. (2001). Co-operative ICT-supported learning. A practical approach. In Keil-Slawik, R., Magenheimer, J. (eds) Informatikunterricht und Medienbildung, INFOS 2001 9.GI Fachtagung Informatik und Schule 17.–20.September 2001 in Paderborn, Proceedings, (pp 47–61).
- [Zi94] Zima, P.V. Die Dekonstruktion, Einführung und Kritik, Tübingen, Basel 1994.