



ELAN II-Partner Universität Hildesheim

Abschlussbericht zum Projekt

„Entwicklung softwareergonomischer Graphischer
Benutzerschnittstellen (GUI) in virtuellen Teams mit
objektorientierten Programmier- und Modellierungskonzepten“

Die Integration von Informatik- und
informationswissenschaftlicher Perspektive

verantwortlich: Prof. Dr. Christa Womser-Hacker
Universität Hildesheim
Institut für Angewandte Sprachwissenschaft
womser@uni-hildesheim.de

Projektlaufzeit: 15.11.2004 - 31.12.2005

Projektmitarbeiter: Ralph Kölle, Dipl.-Inform., wissenschaftlicher Mitarbeiter
Glenn Langemeier, Dipl.-Inform., wissenschaftlicher Mitarbeiter
Steffen Weichert, M.A., wissenschaftlicher Mitarbeiter
Kerstin Bischoff, studentische Hilfskraft

1. Einleitung

Modernisierung und nachhaltige Verbesserung der Hochschullehre durch den Einsatz multimedialer Technologien zur Unterstützung, Erweiterung oder Virtualisierung von Präsenzangeboten nimmt an deutschen Hochschulen einen zunehmend größeren Stellenwert ein. Förderprogramme zum Ausbau der medientechnischen Voraussetzungen an einzelnen Standorten oder das BMBF-Programm „Neue Medien in der Bildung“ entsprechen dieser Entwicklung. Nicht nur im Hinblick auf den Bologna-Prozess, der sich das Zusammenwachsen des europäischen Hochschulbereichs zum Ziel setzt, verspricht dabei der hochschulübergreifende Austausch von Wissen und Erfahrungen zum eLearning sowie die Schaffung einer gemeinsamen Infrastruktur große Mehrwerte.

Daher verfolgt das von 2002 bis 2006 aus Landesmitteln geförderte ELAN, das eLearning Academic Network Niedersachsen, „eine landesweite Vernetzung E-Learning-mitgestaltende(r) Hochschulen unter einer übergreifenden Organisationsstruktur“ [ELAN Niedersachsen]. Dadurch werden Hochschulen und Weiterbildungseinrichtungen in Niedersachsen bei der Planung, Durchführung und Evaluation von multimedialer Lehre unterstützt. Drei Netz-Piloten sollten dazu in einer ersten Förderphase durch die Bereitstellung technischer und organisatorischer Infrastrukturen, Beratungs- und Betreuungsdienstleistungen zum Thema eLearning und beispielhafte Teilprojekte zu Inhalten ein tragfähiges eLearning-Netzwerk etablieren und die Positionierung und Profilierung durch innovative multimediale Lehrangebote forcieren. In einer zweiten Phase wurden zu den Piloten Hannover/Braunschweig, Göttingen/Clausthal und Oldenburg/Osnabrück zusätzlich Partner für die inhaltliche und strukturelle Ergänzung der bis dato bereitgestellten Services und Lehrmodule gefördert. Aufgabe dieser Partner war die Erstellung übertragbarer, landesweit nutzbarer und ELAN-konformer Module, um das multimediale Angebot zu erweitern und um durch Zusammenarbeit und gegenseitigen Content-Austausch über Hochschulgrenzen hinweg die Funktionsfähigkeit des Netzwerks zu steigern. [cf. ELAN Niedersachsen]

Als ein solcher ELAN-Partner der 2. Förderstufe beteiligte sich die Universität Hildesheim in Abstimmung mit dem Piloten Hannover/Braunschweig durch die Entwicklung eines interdisziplinären Moduls zur „Entwicklung softwareergonomischer Graphischer Benutzerschnittstellen (GUI) in virtuellen Teams mit objektorientierten Programmier- und Modellierungskonzepten“. Im Folgenden wird der Verlauf dieses Projekts von der ursprünglichen Idee über das Feinkonzept hin zum Einsatz und der Evaluation im Regelbetrieb beschrieben und ein vorläufiges Resümee gezogen.

2. Projektskizze

Im Wandel der Gesellschaft in eine internationale Informations- und Wissensgesellschaft gewinnt das Wissen um die globale Verarbeitung, Gewinnung, Bewertung, Präsentation, Verteilung und Nutzung von Information immer stärker an Bedeutung. Dabei kommt dem so genannten Endbenutzer und somit auch der Qualität und der Adaptivität von Benutzerschnittstellen in der Mensch-Maschine-Interaktion große Bedeutung zu. An der wissenschaftlichen Auseinandersetzung mit dieser Materie sind verschiedene Disziplinen beteiligt, u.a. die Informatik und die Informationswissenschaft. In der Informatik steht in Programmierpraktika bzw. Übungen als praktische Einführungen in die verschiedenen Programmiersprachen zumeist die Technik im Mittelpunkt. Die Informationswissenschaft fokussiert hingegen auf den Benutzer komplexer Systeme und zielt somit auf die benutzungsgerechte Gestaltung der Schnittstelle zwischen Mensch und Computer. Ziel der Teildisziplin Softwareergonomie ist es dabei, den kognitiven Aufwand für Interaktionen zwischen Mensch und Maschine zu minimieren und so dem Anwender die Konzentration auf inhaltliche Probleme zu ermöglichen. Da Programmierkenntnisse und Softwareergonomie-Know-How derzeit selten in ein und derselben Person vorhanden sind, erscheint es sinnvoll, die Studierenden durch die explizite Verbindung zwischen nötigen Kompetenzbereichen zu unterstützen und Module bereitzustellen, die polyvalent einsetzbar sind und damit verschiedene, aktuelle Fragestellungen bedienen.

Das ursprüngliche Projektvorhaben sah folglich die interdisziplinäre Content-Entwicklung im Schnittbereich zwischen Informationswissenschaft und Informatik vor, welche die beiden Bereiche Softwareergonomie und Objektorientierte Softwareentwicklung synergetisch zusammenführt und um den Aspekt der Multilingualität/Internationalität erweitert. In einer komplexen ELAN-Lerneinheit sollten dazu Kenntnisse aus verschiedenen Bereichen der Mensch-Maschine-Interaktion, des Software Engineering, der objektorientierten Modellierung und Programmierung mit Schwerpunkt auf der Entwicklung softwareergonomischer, graphischer Benutzerschnittstellen (GUIs) vermittelt werden. Aufgrund von Veränderungen, die sich zwischen dem ersten Projektplan im Februar 2003 und dem tatsächlichen Projektbeginn im November 2004 ergeben haben und in Folge finanzieller Kürzungen wurde der Bereich der Multilingualität zunächst verworfen und es erfolgte eine Konzentration auf eine Lerneinheit für virtuelle Teams zur objektorientierten Programmierung.

Zu diesem Zweck wurde die geplante Zusammenarbeit mit dem Projekt VitaminL (Virtuelle Teams: Analyse und Modellierung in netzbasierten Lernumgebungen, <http://www.uni-hildesheim.de/~vitaminl>) [siehe Kölle & Langemeier 2004; Kölle et al. 2005] verstärkt, welches sich im Rahmen mehrerer Promotionsvorhaben mit Rollen in virtuellen Teams

beschäftigt. Kern von VitaminL ist die Analyse von Rollen in einem gegebenen virtuellen Team und das Schaffen einer Möglichkeit zur Kompensation erkannter Defizite eines virtuellen Teams durch geeignete Software-Komponenten (agenten-basierte Moderatoren und Tutoren). Die virtuellen Teams bestehen hier größtenteils aus den – je nach Studiengang – ca. 50 bis 80 Studierenden, die jedes Semester an der Universität Hildesheim an Veranstaltungen zur Einführung in die Programmierung mit der objektorientierten Programmiersprache Java teilnehmen [Kölle et al. 2005]. Ein wichtiger Bestandteil dieser Einführungen sind die von Tutoren betreuten Übungen, so genannte Tutorien, in denen die vermittelten Kenntnisse durch das gemeinsame Lösen von praxisnahen Programmieraufgaben angewandt werden.

Erfahrungen aus vergangenen Programmierkursen zeigen, dass dieses Lernen und Arbeiten in Teams bei hohem Grad an Interaktivität dem Lernerfolg förderlich sind [cf. auch Schulmeister 1997]. Gerade im Hinblick auf individuelle Vorkenntnisse, soziale Kompetenzen und Zielsetzungen heterogenen Gruppen ist ein erhöhter Wissenstransfer innerhalb der Gruppe durch Diskussion zu erwarten, welches wiederum die Reflexion des jeweils eigenen Wissens impliziert [cf. Koelle & Langemeier 2004]. "Interaktivität" setzt hier eine aktive Rolle des Benutzers voraus, die Freiheitsgrade bei der Auswahl zulässt und den Lernweg durch die Lernenden bestimmbar macht. Mit dem gewählten Ansatz verfolgen wir das konstruktivistische Paradigma der Lerntheorie, das Lernen nicht als Instruktionsprozess auffasst, sondern als Konstruktion von Wissen in komplexen Prozessen der Problemlösung [cf. Schulmeister 1997, Klimsa 1993, Mandl et al.1997, Gräsel 1997].

Während viele Ansätze die Übertragung der Theorieteile traditioneller Lehrveranstaltungen zur Programmierung ins eLearning mittels einfacher Hypertext-Tutorials bis hin zu komplexen adaptiven Hyperbook-Systemen verfolgen [Henze et al. 2001], steht hier die Virtualisierung der vertiefenden Praxisteile im Vordergrund. Die Adaption dieser Tutorien auf virtuelle Teams bietet neben räumlicher und zeitlicher Flexibilität auch die Chance, Erfahrungen mit neuen Technologien des CSCL (Computer Supported Collaborative Learning) zu sammeln [cf. Koelle & Langemeier 2004] und dadurch kommunikative und soziale Kompetenzen – Schlüsselqualifikationen für die spätere Berufspraxis – zu erweitern [cf. Womser-Hacker 2004]. Schließlich eröffnet eine CSCL-Umgebung neue Möglichkeiten zur Analyse und Erkennung von Problemsituationen in solchen Teams, so dass die automatisierte Unterstützung durch einen virtuellen Tutor zu einer Entlastung von personalintensiven Tutorien führen würde [cf. Kölle et al. 2005].

Da sich das Projekt mit VitaminL somit auf die Konzeption und Entwicklung von Systemen zur Durchführung dieser Tutorien konzentriert, wird der Content aus vorhandenen Bereichen angepasst bzw. importiert. Dazu wurde in der ersten Phase im Sommersemester 2005 in einer Kooperationsveranstaltung mit der Universität Konstanz der Content teils traditionell

teils per Videokonferenz vermittelt und anschließend per virtueller Tutorien vertieft. An dieser hochschulübergreifenden Kooperation zeigt sich die interdisziplinäre Ausrichtung der Lerneinheit an einer heterogenen Zielgruppe, hier die interdisziplinäre Zusammenarbeit zwischen Informatikern und Informationswissenschaftlern. Sie ist für alle Studiengänge ausgelegt, bei welchen objektorientierte Softwareentwicklung eine Rolle spielt, aber unter Umständen nicht - wie bei Informatik-Studiengängen - im Zentrum steht. Die Teams der Kooperationsveranstaltung bestanden so aus Konstanzer Studierenden des Studiengangs Information Engineering, die größtenteils bereits Programmierkenntnisse besaßen, Studierenden des Studiengangs International Economic Relations und des Studiengangs Biologie sowie den Hildesheimer Studierenden des Studiengangs Internationales Informationsmanagement (IIM) ohne derartige Vorkenntnisse. Ebenso wird das Modul im Studiengang Internationales Informationsmanagement und Informationstechnologie (IM/IT) an der Universität Hildesheim eingesetzt. Die bereits etablierte und bewährte Zusammenarbeit zwischen dem Institut für Physik und Technik und dem Institut für Angewandte Sprachwissenschaft (Schwerpunkt Informationswissenschaft und Internationales Informationsmanagement) wirkt sich somit auf die dynamische Zusammenführung von Angeboten positiv aus und ermöglicht auch eine polyvalente Nutzung interdisziplinärer Bereiche. Das Projekt ist aber zugleich so angelegt, dass es sich problemlos an andere Studiengänge anpassen lässt.

Mit der Integration der Inhalte aus dem Bereich Softwareergonomie und eines virtuellen Labors (Usability Lab) in der zweiten Phase dürfte sich mit den Schlüsselkompetenzen auch die Zielgruppe und damit der Interessentenkreis im Sinne einer zumindest landesweiten Nutzung zusätzlich vergrößern. Hierzu dürften Lernende aus Bereichen, die nicht primär der Informatik zugeordnet sind, z.B. angehende Informationswissenschaftler sowie Kultur-, Medien- und Sprachwissenschaftler, Kognitionswissenschaftler mit entsprechenden Nebenfächern und Vertiefungsfächern, zählen. Auch für die Weiterbildung von Information professionals (Informationsvermittler, Mitarbeiter von IuD-Abteilungen etc.) sind derartige Inhalte von Interesse. Bei der Erstellung und Aufbereitung des softwareergonomischen Contents und der Realisierung des Usability Labs werden didaktisch-pädagogische Aspekte berücksichtigt, wobei besonders die lerntheoretischen Erkenntnisse und Inhalte aus dem Projekt SELIM (Software Ergonomie und Lernen im Multimedialen Kontext, Schudnagis & Womser-Hacker 2002) einfließen werden. Hierzu wird zudem ein Austausch mit dem Kompetenzbereich 3 "Mediendidaktik, Qualitätssicherung und Evaluierung" des ELAN-Pilot Hannover/Braunschweig für sinnvoll und erstrebenswert erachtet.

Die Lerneinheit zeichnet sich somit durch einen hohen interaktiven Anteil (Orientierung am konstruktivistischen Paradigma) und Polyvalenz aus: Eignung für Präsenzlehre oder zum Selbststudium, Kontextbezogenheit bzw. Anwendungsorientierung (Fragestellungen aus der Praxis) sowie Nicht-Informatiker als Zielgruppe (missing link diverser Studiengänge). Um die

Qualität nachhaltig zu sichern, wird nach Ablauf der ersten Projektphase das entwickelte Modul in den Regelbetrieb übernommen und parallel dazu evaluiert sowie weiterentwickelt. Im nächsten Abschnitt wird eine detaillierte Beschreibung des Feinkonzeptes einschließlich der Didaktik, der konkreten Gestaltung der CSCL-Umgebung, der Technik und der Organisation bzw. des Ablaufplans des Projekts gegeben.

3. Feinkonzept

3.1 Didaktik

Entsprechend dem Ansatz des Blended Learning besteht die in der ersten Phase des Projekts umzusetzende Lerneinheit zur objektorientierten Programmierung in Java einerseits aus der theoretischen Vermittlung der Inhalte mittels Vorlesung und andererseits aus den vertiefenden virtuellen Gruppenübungen über die CSCL-Umgebung VitaminL zusammen. In der beschriebenen Kooperationsveranstaltung der Universitäten Konstanz und Hildesheim im Sommersemester 2005 wurden dazu die Vorlesungsinhalte per Video-Konferenz (Polycom Videostation FX) und VNC (Virtual Network Computing) nach Konstanz übertragen.

Die effiziente Verwaltung der Teilnehmer und Gruppen, die Bereitstellung der aufbereiteten Lehrmaterialien und Übungsaufgaben übernahm dabei das in Konstanz im Rahmen des K3-Projektes entwickelte Lernmanagement-System K3-forum [cf. Kölle et al. 2005]. Zusätzlich bietet das System mit Email oder moderierten und unmoderierten Foren Werkzeuge zur asynchronen Kommunikation zwischen Teilnehmern bzw. zwischen Teilnehmern und Dozenten [cf. Kölle et al. 2005]. Für die virtuell in Kleingruppen von drei bis vier Personen zu lösenden Übungsaufgaben wurde das VitaminL-System eingesetzt, welches die verteilte Zusammenarbeit für die fünf aus Konstanzern und Hildesheimern gemischten Gruppen (insgesamt 8 Gruppen mit 34 Teilnehmern) überhaupt erst ermöglichte.

Ähnlich der Präsenztutorien erhalten hier die Kleingruppen von drei bis vier Studierenden Übungsaufgaben im Bearbeitungsumfang von anfangs ca. ein bis zwei Stunden und können durch dieses gemeinsame Problemlösen das notwendige Wissen aufbauen. Dabei ist das Vorgehen der Gruppe analog zu den bekannten Modellen der Softwareentwicklung z.B. dem Wasserfallmodell [Rechenberg & Pomberger 1999] zu sehen. Demnach muss die Aufgabe zunächst gemeinsam analysiert und Lösungsansätze erkannt werden, bevor die Umsetzung in Code und die termingerechte Abgabe der Lösung erfolgt. Bei auftretenden Problemen werden dabei anwesende Tutoren oder Dozenten um Hilfestellung gebeten, welche dann zumeist in Form von Hinweisen, nicht aber einer fertige Lösung gestellt wird [cf. Kölle & Langemeier 2004]. Zum Ende des Semesters mussten die Teams für den Leistungsnachweis bzw. der CreditPoints zudem eigenständig ein größeres Softwareprojekt

entwickeln, wobei sie für den gesamten Entwicklungsprozess ihres Projekts selbst verantwortlich waren.

Das kollaborative Prinzip der gemeinsamen Lösungsfindung durch Experimentieren, Lernen aus Fehlern und Selbstreflexion und vor allem durch intensiven Wissensaustausch im Team ist praxisorientiert und interaktiv und orientiert sich damit am konstruktivistischen Lernansatzes. Gemäß dieser Lerntheorie wird Wissen vom Lernenden auf Basis von früheren Erfahrungen, Vorwissen und sozialem Kontext aktiv und individuell konstruiert. Da daher viele Möglichkeiten der individuellen Wissenskonstruktion existieren und der Lernprozess kaum vorhersagbar ist, soll das Ziel „die Entwicklung von Lernumwelten, in denen kognitive Lernprozesse in handelnder Auseinandersetzung mit der Umwelt stattfinden können,“ [Schulmeister 1997, 78] sein [cf. Kölle & Langemeier 2004]. Die interdisziplinäre Zusammenarbeit zwischen Studierenden verschiedener Studiengänge, wie bspw. in der Kooperationsveranstaltung zwischen IIM, Information Engineering, International Economic Relations und Biologie, sowie unterschiedliche Vorkenntnisse, Zielsetzungen und soziale Kompetenzen fördern dabei Wissenstransfer durch Diskussion und somit die Reflexion des individuellen Wissens [cf. Kölle & Langemeier 2004, Kölle et al. 2005].

Virtuelle Tutorien mit VitaminL

Für die Übertragung dieser Tutorien auf virtuelle Teams bietet die CSCL-Umgebung VitaminL solchen Teams Werkzeuge zur kurzzeitigen, synchronen Bearbeitung von Programmieraufgaben. Das Projekt SELIM, welches in der zweiten Phase zur Erweiterung um softwareergonomische Contents in die Lerneinheit eingebunden werden soll, verfolgt ebenfalls die Virtualisierung von Tutorien, unterstützt allerdings die individuelle Lernform. Für die gemeinsame verteilte Programmierung setzt VitaminL eine integrierte Entwicklungsumgebung (Integrated Development Environment; IDE) um, die die gängigen Werkzeuge der Java-Programmierung (Compiler, Interpreter) und der synchronen Kommunikation zur Verfügung stellt. [cf. Kölle et al. 2005].

Da die Teams – als in der Regel Programmieranfänger – bei der Aufgabenlösung immer wieder auf Probleme verschiedener Art stoßen werden, nehmen die Tutoren eine zentrale Stellung ein. Problemsituationen können aus Verständnisproblemen bei der Aufgabenstellung, aus der Syntax und Semantik der Sprache oder aus dem Einsatz algorithmischer Strukturen resultieren [cf. Kölle et al. 2005]. Um die virtuellen Teams dabei zu unterstützen, wurden sie in der erwähnten Kooperationsveranstaltung von einem in Hildesheim anwesenden Tutor betreut. Auch konnten außerhalb der regulären Vorlesungs- bzw. Übungszeiten Termine für ein virtuelles Treffen mit dem Tutor vereinbart werden.

Langfristig jedoch verfolgt das Projekt VitaminL die automatische Unterstützung in Problemsituationen, um auch bei Nicht-Verfügbarkeit eines realen Tutors Hilfe anbieten und

damit Frustration vermeiden zu können. Während sich technische Probleme wie die Installation der IDE und der Javakomponenten durch schriftliche Anweisungen und organisatorische Probleme wie die Abstimmung von Terminen durch Integration von Groupware-Werkzeugen leicht bewältigen lassen, ist das Erkennen und Beheben inhaltlicher Schwierigkeiten sehr komplex [cf. Kölle et al. 2005].

Um in derartigen Problemsituationen die virtuelle Lerngruppe adäquat unterstützen zu können, werden Techniken benötigt, die von intelligenten tutoriellen Systemen (ITS) eingesetzt werden. So beobachten Systeme wie bspw. COMET, Algebra-JAM oder C-CHENE die Teilnehmer während der Kollaboration, ziehen davon ausgehend Rückschlüsse auf aktuelle Problemsituationen oder auf die Zusammensetzung des Teams (auf der Grundlage eines geeigneten Rollenmodells) und reagieren entsprechend den ermittelten Daten mit tutoriellem Support. Die geplante tutorielle Komponente von VitaminL basiert auf der Annahme, dass eine effektive Zusammenarbeit von der personellen Zusammensetzung des Teams entscheidend beeinflusst wird. Unter der Annahme, dass viele Problemsituationen in Folge eines Defizits an für die Zielerreichung wichtigen Funktionen in einem Team entsteht, wurde ein speziell auf den Anwendungsbereich adaptiertes Rollenmodell entwickelt, auf dessen Basis eine gezielte Unterstützung erfolgen soll. [cf. Kölle et al. 2005, Kölle & Langemeier 2004]

Dazu werden allgemeine Modelle zu Arbeitsfunktionen [Margerison 1990; TMS-Zentrum 2003] auf die in der objektorientierten Software-Entwicklung typischen Tätigkeiten, wie sie gängige Vorgehensmodellen (Wasserfallmodell [cf. Rechenberg & Pomberger 1999] oder dem V-Modell [IABG 2004]) beschreiben, angepasst und mit den Persönlichkeitstypen nach Spencer & Pruss [1995] zusammengeführt. Das Ergebnis sind zehn Rollen, die sich nach funktionalen, sozialen und integrierenden Rollen klassifizieren lassen.

Mithilfe einer strukturierten Kommunikationsschnittstelle in der CSCL-Umgebung sollen die kommunikativen Tätigkeiten gemäß den *Collaborative Learning Skills* [McManus & Aiken 1995] kategorisiert werden, so dass jedem Persönlichkeitstypen typische Kommunikationsakte zugeordnet werden können, die später zur Identifizierung dienen. Eine Softwarekomponente analysiert auf der Basis dieser Daten das Team während der Zusammenarbeit und sendet Rollenprofile und Kennzahlen wie z.B. Bearbeitungs-, Kommunikations- und Navigationsanteile jeden Benutzers und der gesamten Gruppe mit der Information, ob eine Problemsituation vorliegt oder nicht, an eine Simulationskomponente, die im Falle einer erkannten Problemsituation das Team mit geeigneten Maßnahmen im Rahmen detektierter Rollendefizite unterstützt. Um weder den Lerneffekt zu verhindern noch die Motivation zu untergraben, soll dabei nur in Situationen, in denen sich das Team aufgrund fehlender Rollen nicht selbst helfen kann, durch Simulation der Funktionen

entsprechenden Rolle eingegriffen werden. Dabei soll die Hilfe kommunikativ, nicht jedoch in Form von Eingriffen in die Codierung des Quelltextes erfolgen [Kölle & Langemeier 2004].

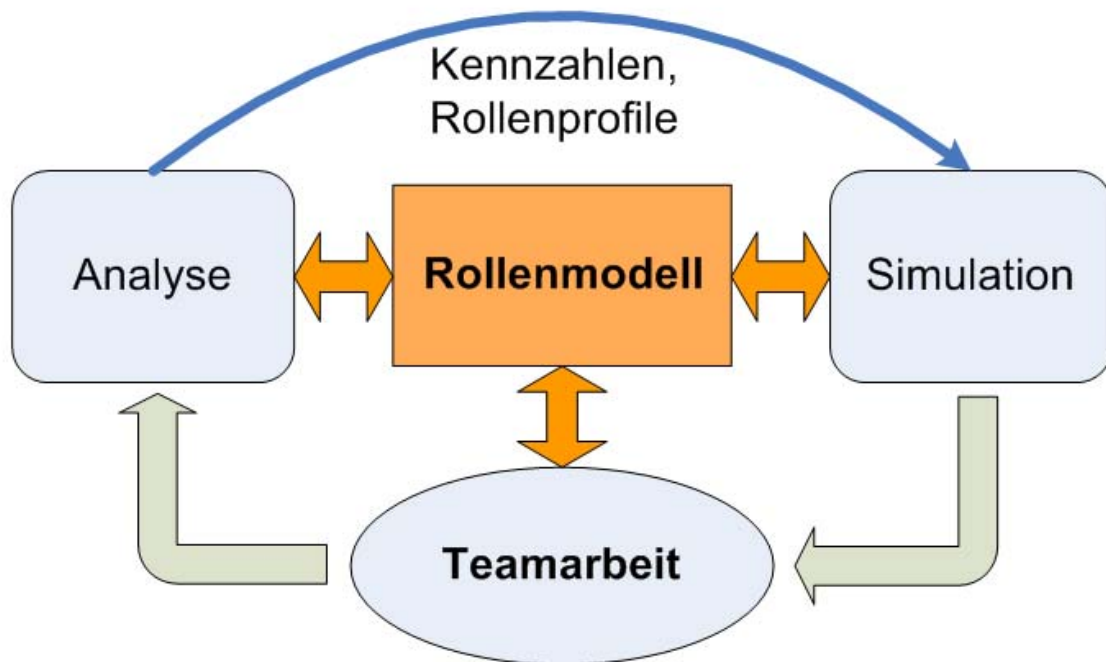


Abbildung 1: Analyse und Simulation auf Basis des Rollenmodells

Die Realisierung dieser automatischen tutoriellen Komponente weist jedoch über den Rahmen der hier geplanten Lerneinheit im Rahmen von ELAN hinaus, so dass die folgende Beschreibung der Gestaltung und Technik der Lernumgebung sich auf den Stand zur Durchführung der Kooperationsveranstaltung im Sommersemester 2005 bezieht. Da die Erweiterung des Projekts um den softwareergonomischen Content erst in einer zweiten Phase geplant war, wird zugunsten eines abschließenden Ausblicks zum Gesamtprojekt an dieser Stelle auf eine genaue Betrachtung verzichtet.

3.2 Technik und Lernumgebung

Da die vom Piloten Hannover/Braunschweig im Kompetenzbereich 1 "Technik und Infrastruktur" zur Verfügung gestellten Lernplattformen (Clix, Blackboard und Hyperwave) sich für den Anwendungsfall einer verteilten kollaborativen Programmierplattform als ungeeignet erwiesen, sollte für die geplante CSCL-Umgebung ein erster Prototyp der VitaminL-Plattform für den reellen Einsatz weiterentwickelt werden. Nachdem durch Benutzertests und Fragebögen ermittelte Fehler und konzeptionelle Schwächen unter anderem durch Re-Design behoben worden waren, entstand die in der Kooperationsveranstaltung im Sommersemester 2005 eingesetzte Version von VitaminL. Abbildung 1 zeigt die Benutzeroberfläche des VitaminL-Clients.

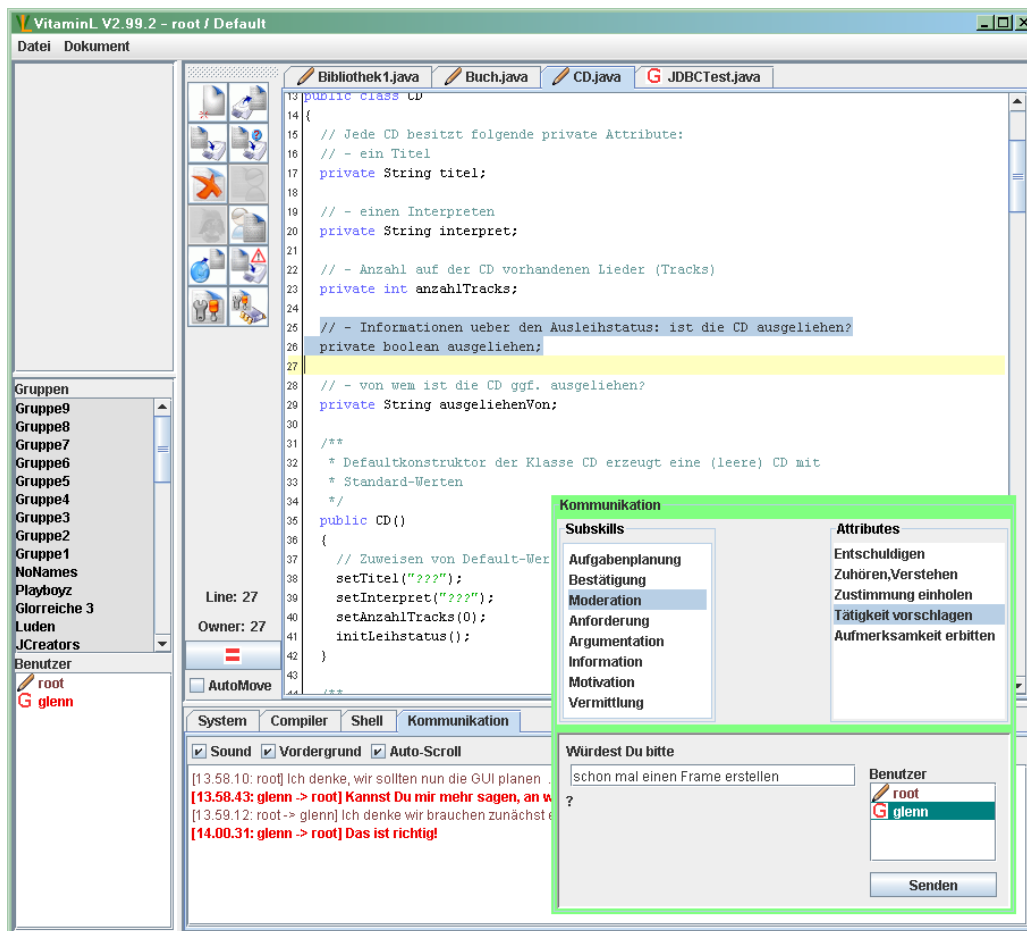


Abbildung 2: Screenshot des VitaminL-Clients V2.99.2

Unter der Menüzeile mit den wichtigsten Kommandos werden im linken Bereich die derzeit verfügbaren Gruppen angezeigt, wobei aktive Gruppen weiß hinterlegt sind. Darunter ist sichtbar, welche Benutzer eines Team aktuell angemeldet sind. Der Hauptarbeitsbereich, *Shared Documents*, dient der gemeinsamen Bearbeitung von Dokumenten. Dazu dient die Übersicht über alle derzeit von Mitgliedern der Gruppe geöffneten Dokumente und die Werkzeuge zum Anlegen, Öffnen, Speichern, Schließen, Kompilieren und Ausführen der Dokumente. Mitglieder eines Teams können so die Programmierung der anderen Mitglieder verfolgen und zudem Dokumente untereinander austauschen, um einen Quelltext gemeinsam zu entwickeln.

Den unteren Teil nimmt ein Informationsfenster ein, in dem Systemmeldungen, Compilermeldungen, die Ein- und Ausgaben während der Programmausführung sowie alle in der Gruppe verschickten Kommunikationsnachrichten erscheinen. Aktiv kommuniziert wird dabei mittels eines frei verschiebbaren Kommunikationsfensters, welches entweder als Freichat oder als strukturierte Dialogschnittstelle vorgegeben wird. In der chat-basierten Variante wird eine Nachricht in ein Textfeld eingegeben und an die gesamte Gruppe versendet oder es wird ein dezidiertes Empfänger ausgewählt, in dessen Oberfläche diese Nachricht dann hervorgehoben dargestellt wird. Gleichzeitig kann sich ein Benutzer durch ein akustisches Signal über den Empfang einer an ihn gerichteten Nachricht informieren

lassen. In der strukturierten Form der Kommunikation werden statt einer Freitexteingabe gemäß dem kommunikationstheoretischen Ansatz der *collaborative learning skills* [McManus & Aiken 1995] Kategorien und ihnen zugeordnete Satzanfänge ausgewählt, die dann sinngemäß vervollständigt werden.

Diese anfangs ungewohnte Form, Nachrichten zu schreiben und zu versenden, bietet nach einer Einarbeitungszeit neben einer Reduzierung der Schreiarbeit auch eine gezieltere und effektivere Kommunikation. Diese wurde in einigen Benutzertests eingesetzt, jedoch nicht in der Kooperationsveranstaltung, um den Teilnehmern den Einstieg in das Programmieren lernen über eine CSCL-Umgebung zu erleichtern. Generell jedoch ist die strukturierte Kommunikation die Grundlage des Arbeitsmodells von VitaminL, da über Sie die Rollenbesetzung eines Teams bestimmt und damit die adäquate tutorielle Unterstützung erfolgen soll. Die Unterstützung der Gruppen in der Kooperationsveranstaltung war vorerst durch einen Tutor in Hildesheim wahrzunehmen, der sich als Benutzer in die verschiedenen Teams einloggen konnte und mit der gleichen Funktionalität ausgestattet durch Einsehen aller Dokumente und der Kommunikation Hilfestellung leisten konnte. Für die Installation des JDK und der VitaminL-Software, letztere über eine ausführbare JAR-Datei, wurde den Teilnehmern über das K3-Forum eine Dokumentation bereitgestellt und bei Schwierigkeiten halfen Tutor und Dozenten.

Die VitaminL-Plattform ist eine vollständig in Java realisierte Eigenentwicklung der Autoren [siehe Kölle & Langemeier 2004]. Sie ist als Client-Server-Architektur konzipiert, wobei der Datenaustausch zwischen Server und Clients ursprünglich durch Remote Method Invocation (RMI) über TCP/IP-basierte Netzwerkverbindungen geplant war. Für den standortübergreifenden Einsatz in der Kooperationsveranstaltung wurde das Netzwerkprotokoll durch HTTP-basierte Verbindungen ersetzt, um eine einfache Installation für die Benutzer zu ermöglichen und Probleme durch Firewalls zu umgehen [cf. Kölle et al. 2005]. Alle Aktionen einer Teamsitzung, sowohl in der Kommunikation als auch bei der gemeinsamen Dokumentenbearbeitung, werden auf diese Weise an den Server geschickt und von dort entsprechend weitergeleitet. Durch die Zentralisierung ist es möglich, die gesamte Kommunikation zu erfassen und darüber später die Teamunterstützung zur Verfügung zu stellen. Gleichzeitig erlauben Protokolle der Sitzungen in Form von Logfiles¹ durch ihre nachträgliche Rekonstruktion mittels eines Logfileanalyzers die Evaluierung der Plattform und die Analyse von Problemen bei der Zusammenarbeit. Da die Architektur auf einfachen Schnittstellen und einem transparenten Nachrichtenprotokoll beruht, wird die Erweiterung des Systems auch durch externe Entwickler vereinfacht, so dass im Rahmen von Projekt- oder Abschlussarbeiten eine Weiterentwicklung denkbar ist. [cf. Kölle & Langemeier 2004]

¹ Die Teilnehmer von Benutzertests sowie jene der Kooperationsveranstaltung wurden über die Logfileerfassung ihrer Zusammenarbeit sowie die damit verbundenen Ziele informiert und gaben ihre Zustimmung.

3.3 Projektplan

Zur Umsetzung der geplanten Lerneinheit waren zu Projektbeginn folgende Arbeitspakete mit den jeweils zu erwartenden Mitarbeitermonaten veranschlagt:

- Analyse des State of the Art: GUI-Entwicklung, CSCL-Umgebungen (1MM)
- Aufbau der Serverkomponenten, Versionsverwaltung, CSCL (1MM) inklusive Überprüfung der Ergebnisse von ELAN-Piloten hinsichtlich von CSCL-Umgebungen auf Eignung
- Anpassen der ausgewählten CSCL-Umgebung an die eigenen Bedürfnisse (1MM)
- Analyse, Modellierung und Design des 1.Prototypen (2MM)
- Erstellung des 1. Prototypen, Entwicklung neuer und Integration vorhandener Materialien (1,5MM)
- Durchführung der 1. Testphase (4MM)
- Auswertung der 1. Testphase, Anpassung und Optimierung der Arbeitspakete auf Basis des Ergebnisses (3,5MM)
- Durchführung 2. Testphase (4MM)
- Durchführung der Maßnahmen zum Einsatz im Regelbetrieb, beinhaltet u.a. Dokumentation und Projektbericht (3MM)

Bei dieser Detailplanung ist zu berücksichtigen, dass die Phase bis zum 1.Test nach sechs Monaten abgeschlossen sein musste, um in der Kooperationsveranstaltung mit Konstanz den Prototyp einsetzen und evaluieren zu können. Daraus ergaben sich als Meilensteine:

Meilensteine

- 15.11.2004 Beginn des Projekts
- 01.04.2005 1.Testphase: Einsatz Prototyp im Sommersemester 2005
- 01.10.2005 2.Testphase: Einsatz des weiterentwickelten Prototyps mit eingeflossenen Ergebnissen der 1.Testphase
- 01.03.2006 Auswertung der 2.Testphase und Anpassungen und Optimierungen aufgrund der gewonnenen Ergebnisse
- 01.04.2006 Übergang in den Regellehrbetrieb

Jahr	2004			2005									2006											
Monat	10	11	12	01	02	03	04	05	06	07	08	09	10	11	12	01	02	03	04	05	06	07	08	...
Ziele	Analyse und Design, Entwicklung Prototyp			Testphase I Durchgang Sommersemester					Testphase II (Auswertung, Optimierung der Testphase I) Durchgang Wintersemester				Abschluss- / Optimierungs - arbeiten Übergang in den Regelbetrieb											

Abbildung 2: Zeitplan

Im Detail sollte nach der Analyse des State of the Art, der Sichtung der ELAN-Piloten auf verwendbare CSCL-Umgebungen, dem Aufbau der Serverkomponenten und dem Anpassen einer geeigneten CSCL-Umgebung, parallel zur Analyse und Erstellung der software-ergonomischen Materialien im Sommersemester eine umfangreiche Evaluierung stattfinden. Neben der Kooperationsveranstaltung waren dazu umfangreiche Benutzertests und Fragebögen sowie die Vergabe von Abschlussarbeiten zu spezifischen Themen wie Problemsituationen während der Programmierung und in der Kommunikation geplant. Dazu bot sich besonders die Auswertung der entsprechend aufgezeichneten Logfiles an, um auf Basis der Ergebnisse auch im Hinblick auf Möglichkeiten zur automatischen Unterstützung durch eine Tutorenkomponente in der zweiten Testphase einen optimierten Prototyp einsetzen zu können. Evaluierende Maßnahmen begleiteten bereits die Entwicklung des ersten Prototyps, der im Anschluss an Vorarbeiten im Rahmen des Projekts VitaminL unter stetigen Benutzertests und einer Pilotstudie entstand.

Um die anfallenden Aufgaben zur Analyse, Aufbereitung, Modellierung und Realisierung sowie die Durchführung der Kurse und zusätzlicher Benutzertests bearbeiten zu können, wurde ein wissenschaftlicher Mitarbeiter für eine Laufzeit von 14 Monaten angestellt und die Mitarbeiter des Promotionsprojekts VitaminL eingebunden. Zusätzlich wurde eine studentische Hilfskraft mit 30 h pro Monat eingesetzt, um bei Literaturrecherche, Implementierung, Testen von Software und im Rahmen von Benutzertests zu unterstützen. Auch wenn es sich bei der Kooperationsveranstaltung in der Testphase des Prototypen um eine Exportveranstaltung handelte, übernahmen die Konstanzer Partner während des Kurses Aufgaben zur Verwaltung der Gruppen und der in Hildesheim erstellten Lehr- und Übungsmaterialien über das K3-System.

4. Projektverlauf und Evaluation

4.1 Entwicklung der CSCL-Umgebung

Nachdem sich weder in der Sichtung der Ergebnisse der ELAN-Piloten und den von ihnen zur Verfügung gestellten Lernplattformen noch in der Analyse des State of the Art eine Umgebung für das gemeinsame Lernen einer Programmiersprache als geeignet erwies, wurde an die Forschungsarbeiten des Projekts VitaminL angeknüpft. Der Prototyp, für dessen Entwicklung die Vorarbeiten bereits seit Stellung des ursprünglichen Projektantrags (im Jahre 2002/2003) liefen, wurde dazu Benutzertests unterzogen. In einer Pilotstudie wurde gleichzeitig der Teamfragebogen nach Spencer & Pruss [1995] elektronisch umgesetzt und damit Daten erhoben, um das Grundidee und Fernziel der rollenbasierten Unterstützung zu überprüfen [cf. Kölle & Langemeier 2004].

Die Beantwortung des Fragebogens ergibt dabei für jeden Befragten Punktwerte, die Aufschluss über Neigungen zu jeder der vorhandenen Rollen geben. Die Auswertung der Fragebögen zeigte, dass bei Zusammenführen der einzelnen Profile zu einem Gruppenprofil, häufig bestimmte Rollen, z.B. die des Informationsbeschaffers, über- oder unterrepräsentiert sind. Daher liegt der Schluss nahe, bereits existierende CSCL-Techniken um die zielgerichtete Unterstützung von Teams unter Berücksichtigung ihrer jeweiligen Teamzusammensetzung zu ergänzen [cf. Kölle & Langemeier 2004].

Um den entsprechend konzipierten Prototypen für die Virtualisierung von Programmierkursen in einer verteilten Zusammenarbeit via Internet und mit mehreren Gruppen einsetzen zu können, wurden vor der ersten Testphase mittels einiger Benutzertests Schwächen und Fehler behoben. In diesen Benutzertests bearbeiten Teams aus zwei bis vier Studierenden verteilt auf verschiedene Räume kleinere Aufgaben, wobei sie Bücher, Skripte und das Internet für Hilfestellungen nutzen können. Kommuniziert und programmiert wurde dabei ausschließlich über die CSCL-Umgebung des Projekts. Die Hauptkritik, dass der Prototyp nur einem Mitglied die eigentliche Programmierung erlaubt, da lediglich ein Dokument bearbeitet werden konnte, wurde durch die Verwendung eines *Multi Document Editors* beseitigt [cf. Kölle & Langemeier 2004]. Zudem ergaben die Tests, dass im Editor für das gemeinsame Programmieren wichtige Funktionen vermisst wurden.

Da die nötige Umstellung der Netzwerkverbindungen auf HTTP-basierte Verfahren und der Aufbau der Server für die synchrone Anmeldung mehrerer Gruppen und Teilnehmer zu Beginn der Testphase noch größere technische Schwierigkeiten bereitete, waren noch nicht alle geplanten Änderungen zu Kursbeginn integriert. Zusammen mit dem Feedback der Teams in der Kooperationsveranstaltung der Testphase wurde der Funktionsumfang nach und nach ausgebaut. Der Editor wurde um *Syntax Highlighting* und grundlegende Awareness-Funktionen erweitert. Zeilennummern für Java-Quelltexte sowie die Markierung der aktuell von einem Benutzer bearbeiteten Zeile wurden eingeführt, so dass von allen Teammitgliedern sofort nachvollzogen werden kann, wo gerade codiert wird. Ein zusätzliches Navigationselement für die Beobachter zum schnellen Sprung zur vom Besitzer bearbeiteten Zeile wurde ergänzt. Über das Setzen von Markierungen im Quelltext, die an das Team übermittelt werden, ist es dem Besitzer eines Dokuments leicht möglich, den anderen etwas zu erklären, ohne kompliziert auf die entsprechende Stelle über Kommunikationsnachrichten verweisen zu müssen. Des Weiteren wurde eine ‚Undo/Redo‘-Funktion und die Ausführung von Java-Programmen mittels Java-Interpreter direkt innerhalb der IDE integriert. Infolge der Serverprobleme zu Beginn des Kurses zeigte sich die Bedeutung von lokalen Kopien der geöffneten Dokumente, die ein Client bei Wegfall der Netzwerkverbindung abspeichern kann [cf. Kölle et al. 2005].

Der Darstellung der Änderungen an der CSCL-Umgebung, die auf der empirischen Evaluation durch regelmäßige Benutzertests und den Einsatz in der Kooperationsveranstaltung beruhen, folgen nun didaktische Betrachtungen zum Verlauf des Kurses.

4.2 Evaluation in Kooperationsveranstaltung

Das Interesse, an einer Lehrveranstaltung dieser Art teilzunehmen und damit Erfahrungen mit Techniken der virtuellen Lehre und CSCL-Umgebungen zu sammeln, war von Seiten der Studierenden groß. Auch gab es in Konstanz bis dato keine Lehrveranstaltung im Bereich Java und aufgrund der Popularität in Wirtschaft und Wissenschaft war die Nachfrage so stark, dass nach Ankündigung des Kurses die maximale Teilnehmerzahl von 35 schnell erreicht war.

Für die in der Präsenzphase nach Konstanz übertragenen Vorlesungsinhalte wurden etablierte Verfahren wie Video-Konferenz (Polycom Videostation FX) und VNC (Virtual Network Computing) verwendet, welche problemlos funktionierten. Erfahrungen aus den in den letzten Jahren in der Informationswissenschaft realisierten Projekten MEUM und Virtueller Campus waren den Dozenten dabei sehr von Nutzen. Dabei konnte kein genereller Mehraufwand der virtuellen Veranstaltung im Vergleich zu traditionellen festgestellt werden. Die Verwaltung der Teilnehmer, der Lehr- und Übungsmaterialien wurde durch das Lernmanagement-System (LMS) in Konstanz entwickelte K3-forum effizient und flexibel wahrgenommen. Die asynchrone Kommunikation via Email und Foren konnte ohne größeren Einarbeitungsaufwand ebenso leicht von den mit dem System nicht vertrauten Hildesheimer Studierenden genutzt werden und diente neben der Abstimmung in und mit den Gruppen vor allem dem Feedback an die Dozenten zur CSCL-Plattform.

Die virtuelle Zusammenarbeit innerhalb der größtenteils zwischen Konstanz und Hildesheim verteilten Teams zur Bearbeitung der Übungsaufgaben bereitete zu Beginn wie bereits erwähnt einige technische Schwierigkeiten. Dabei zeigte sich, dass derartige Probleme die Akzeptanz eines Systems und die Wahrnehmung eines potentiellen Mehrwerts stark behindern können. Damit die Motivation, das System zu nutzen, nicht durch den Einarbeitungsaufwand in neuartige Bedienkonzepte zusätzlich sinkt, ist es entscheidend, den Benutzern Sinn und Vorteil dieser Werkzeuge zu vermitteln. Nach Behebung der anfänglichen Probleme und Erweiterung des Funktionsumfangs besonders des Editors stieß die VitaminL-Plattform zunehmend auf Akzeptanz und konnte durch die dargebotene Möglichkeit überzeugen, flexibel über Ortsgrenzen hinweg gemeinsam zu Programmieren.

Während die Teilnehmer aus Hildesheim als Studierende des Internationalen Informationsmanagements (IIM) zusammen mit einigen Konstanzer Studierenden der International Economic Relations (MA) und der Biologie noch keinerlei Programmierkenntnisse besaßen, handelte es sich bei den Konstanzer Teilnehmern aus dem Studiengang Information Engineering (BA und MA) nicht um Programmieranfänger.

Zunächst führten diese infolge der interdisziplinären Ausrichtung sehr unterschiedlichen Vorkenntnisse bei den Teilnehmern ohne Programmiererfahrung anscheinend zu einem Motivationsverlust. Im Verlauf der Veranstaltung schienen jedoch gerade diese heterogenen Gruppen einen höheren Lernerfolg als in der traditionellen Veranstaltung zu fördern. Durch Wissenstransfer innerhalb des Teams, der sich in der Kommunikation in den Logfiles widerspiegelt, scheinen die Anfänger an dem Wissen ihrer virtuellen Teampartner zu partizipieren. Durch konstruktiven Austausch konnten so zeitnah inhaltliche Fragen und Verständnisprobleme innerhalb der Gruppe geklärt werden. Hingegen sind die Teilnehmer der parallel stattfindenden traditionellen Veranstaltung als elementare Programmieranfänger in solchen Fällen auf aufwendige und zum Teil sicher frustrierende Recherche oder gar auf Hilfe von einem Tutor oder Dozenten angewiesen. Mithilfe der Logfiles lässt sich für einige Gruppen sehr schön rekonstruieren, wie ‚schwächere‘ Teilnehmer systematisch eingebunden und bei ihren Versuchen unterstützt wurden. [cf. Kölle et al. 2005]

Umfangreiche und anspruchsvolle Abschlussprojekte zu Ende des Semesters und die entsprechend höhere Durchschnittsnote bestätigten den Eindruck. Nicht alle der Teams blieben dabei für die Projekte in ihrer ursprünglichen Zusammensetzung, da deren Bearbeitung nur für den Schweinerwerb in Hildesheim nötig war. Ergänzende Regelungen zur Studien- und Prüfungsordnung im Fachbereich Informations- und Kommunikationswissenschaften wurden dazu mit dem Ziel einer nachhaltigen Verwendung multimedial gestützter Anteile in Lehre und Studium frühzeitig getroffen. Einige O-Töne aus der Evaluierung der Kooperationsveranstaltung Hildesheim/Konstanz sollen abschließend das Bild der aus Teilnehmer- und Dozentsicht insgesamt als sehr interessant und viel versprechend einzustufenden Veranstaltung abrunden:

- Studentin:** Mich hat es beeindruckt, dass solche Projekte auch auf so eine große Distanz hin gut zu bewältigen sind.
- Studentin:** Es war gut, dass es auf so große Entfernung möglich war, in einer Gruppe zu arbeiten.
- Student:** Man konnte direkt sehen, was meine Gegenüber programmiert haben und per Chat Einfluss nehmen.
- Student:** Mir hat es gefallen, dass wir alle unser Wissen kombinieren konnten.
- Studentin:** Mich hat es manchmal gestört, dass der persönliche Kontakt gefehlt hat und dass man zuverlässige Gruppenmitglieder braucht, um alles gut organisieren zu können – das war leider nicht immer der Fall.
- Dozent:** Die interaktiven Komponenten der Gruppenarbeit wurden gut angenommen. Manchmal scheint es an Support zu fehlen, so dass die Zusammenarbeit ins Stocken gerät. Die Mitglieder zögern häufig, um Hilfe zu bitten. Es sollte jederzeit ein Tutor zur Verfügung stehen - sei er real oder virtuell - der die Gruppen bei Problemen unterstützt.

Um derartige Situationen, die eine Zusammenarbeit ins Stocken bringen, modellieren zu können, fanden zu der Evaluierung in dieser Kooperationsveranstaltung außerdem im Juli

2005 weitere Benutzertests mit 32 Teilnehmern aus der traditionellen Veranstaltung statt. Zusammen mit den Logfiles aus dem virtuellen Kurs wurden diese intellektuell auf inhaltliche und kommunikative Problemsituationen während der Zusammenarbeit ausgewertet.

4.3 Auswertung der Logfiles

Um eine der zentralen Fragen des Forschungsprojekts VitaminL nach typischen Problemsituationen von Programmieranfängern untersuchen zu können und damit potentielle Unterstützungsmöglichkeiten auf ihre Angemessenheit zu prüfen, waren weitere Benutzertests nötig. Da die Teilnehmer der virtuellen Veranstaltung teils große Vorkenntnisse aufwiesen, zeigte sich in einer ersten Durchsicht der Logfiles, dass hier vermutlich unrepräsentativ wenige fachliche Schwierigkeiten auftraten. In den Benutzertests, in denen je zwei bis vier Studierende des Studiengangs IIM, die parallel die Vorlesung zur Einführung in Java mit Begleitübung absolvierten, eine Aufgabe bearbeiteten, sollten zusätzliche Daten erhoben werden.

Logfiles aus beiden Quellen wurden nachträglich mithilfe eines eigenentwickelten Logfile-Analysers ‚abgespielt‘ und analysiert. Dabei zeigte sich das Arbeitsmodell von VitaminL, welches die Simulation einer im Team defizitären Rolle anstrebt, als eine gute Grundlage. So lassen sich Trends hin zu verzichtbaren und wichtigeren Rollen identifizieren, die selten im angemessenen Maße ausgeprägt bzw. verteilt sind. Ebenfalls anhand der Logfiledaten untersuchte eine im Umfeld des Projekts angelegte Magisterarbeit [Göldner 2005] qualitativ die Kommunikation während der gemeinsamen Programmierung und die damit verbundenen Problemsituationen der Chat-Kommunikation.

Durch die explorative Suche nach typischen inhaltlichen Problemen u. a. durch eine angegliederte Abschlussarbeit im Studiengang IIM [Bischoff 2005] konnte eine Übersicht über prominente Anfängerfehler und Stocksituationen gewonnen werden. Anfangs handelt es sich dabei häufig um einfache Syntaxfehler, wie das Vergessen von Semikolon oder Klammern, später kommen semantische Fehler besonders im Bereich von Arrays und deren Abbruchbedingungen in Iterationen sowie mit Datentypen und ihrer Kompatibilität hinzu. Auch die objektorientierten Konzepte mit Vererbung, der Instanziierung von Objekten und der Verwendung ihrer Attribute stellen zumeist Hürden dar [cf. Kölle & Langemeier 2004, Bischoff 2005].

Es zeigt sich jedoch auch, dass die übermäßigen Syntaxfehler durch Kommunikation im Team oder/ und mithilfe des Compilers schnell gefunden und behoben wurden. Da viele Gruppen eher selten kompilieren könnten hier Hinweise auf bessere Praktiken hilfreich sein. Kaum alleine zu bewältigen waren gerade für die Hildesheimer Anfängergruppen Probleme bezüglich der Zerlegung einer als Text vorgegebenen Aufgabe in algorithmischen Strukturen und deren effektive Übersetzung in die entsprechenden Codefragmente. Im Prozess der Lösungsfindung werden dabei teils nach Versuch-und-Irrtum vielfältige Fehler in den Code

eingefügt oder es wird aus früheren Beispielen kopiert, ohne die nötigen Anpassungen vorzunehmen. Aufgrund der Beobachtung, dass Beispiele in Problemsituationen häufig inadäquat genutzt werden, analoge Beispiele infolge oberflächlicher syntaktischer Merkmale wiederum teilweise nicht erkannt werden, führte zu der Idee einer beispielbasierten Unterstützung [cf. Bischoff 2005] .

Durch das automatische Bereitstellen von geeigneten Beispielen, so die Vermutung, könnten verschiedene Fehlertypen abgedeckt werden und die Aufgabe der häufig wenig ausgeprägten Entdeckerrolle simuliert werden. Im Rahmen der Dissertationsvorhaben in dem Projekt VitaminL ist ein Prototyp für solch eine automatische Simulation in Entwicklung, nachdem erste Benutzertests, in denen in Problemsituationen Beispiele dargeboten wurden, auf eine generelle Tauglichkeit des Ansatzes deuten. Neben einer sinnvollen Kommunikation der tutoriellen Komponente und ihrer Reaktion nicht nur anhand des Quelltextes, sondern auch anhand der (möglicherweise stockenden) Kommunikation innerhalb des Teams, ist hier eine langfristige experimentelle Evaluation von zentraler Bedeutung. So betrachteten die Versuchspersonen Beispiele als eine geeignete Strategie, die sie selbst regelmäßig einsetzten, um Probleme zu überwinden. Art, Elaboration und Umfang der Beispiele sind bei Umsetzung einer solchen Komponente wohl aber eben so kritische Parameter wie die Entscheidung darüber, wann und wie häufig eingegriffen werden soll.

Wie bereits die VitaminL-IDE selbst wird eine zu realisierende Tutorkomponente – und die darin enthaltene Analysekomponente – einer qualitativen Überprüfung unterzogen. Weitere Benutzertests (im Wintersemester 2005/2006 und voraussichtlich im Sommersemester 2006) im Rahmen von an der Universität Hildesheim stattfindenden Java-Veranstaltungen sind dazu geplant und sollen sowohl der Qualitätsüberprüfung der Analyseverfahren als auch ihrer Qualitätssteigerung dienen.

5. Resümee

Der kombinierte Einsatz der CSCL-Plattform VitaminL als synchrones Programmierwerkzeug mit dem K3-Forum für die asynchrone Kommunikation, die Verwaltung der Gruppen und der Lehrmaterialien in einer hochschulübergreifenden Kooperation hat sich als sehr effizient erwiesen. Für beide Partner wurden so verschiedene Mehrwerte geschaffen, da die Konstanzer Teilnehmer einen bis dato nicht angebotenen Kurs belegen und die Teilnehmer in Hildesheim als Folge der interdisziplinären Ausrichtung der virtuellen Teams einen gesteigerten Lernerfolg erzielen konnten. Daher ist eine weitere Zusammenarbeit bei gleichzeitiger Weiterentwicklung der jeweiligen Plattformen durchaus anzustreben. Das K3-Forum wird derzeit um Module zur automatischen Bewertung der Mitarbeit der Studierenden im System erweitert [cf. Kölle et al. 2005]. Bereits im Wintersemester 2005/2006 wurde die

Kooperation zwischen den Universitäten in Hildesheim und Konstanz durch Import einer Vorlesung zu Organisations- und Geschäftsmodellen des elektronischen Publizierens fortgesetzt, wofür wiederum Videokonferenztechniken und das K3-Forum genutzt wurden.

Das VitaminL-System, welches nach anfänglichen technischen Schwierigkeiten zunehmend die Akzeptanz der Benutzer fand, hat durch die stetige Evaluation einen stabilen Stand erreicht und wird durch die Umsetzung einer rollenbasierten Tutorenkomponente zu einem ITS weiterentwickelt werden. Seit der zweiten Testphase wird dazu auf Basis der zuvor ermittelten Ergebnisse die Möglichkeit der Realisierung einer beispielbasierten Unterstützung geprüft. Eine Herausforderung dürfte dabei die zuverlässige und zeitnahe Erkennung der Benutzer- bzw. Teamprofile darstellen, da mit der VitaminL-IDE in den Teamsitzungen zumeist nur Übungsaufgaben im Umfang von bis zu zwei Stunden bearbeitet werden. Gezielte Unterstützung bedeutet Hilfe auf Basis der Teamprofile, denn nur bei Defiziten sollte eingegriffen werden, um die Motivation nicht zu untergraben [cf. Kölle et al. 2005].

6. Ausblick

Beruhend auf den Evaluationsergebnissen ist des Weiteren die Integration zusätzlicher Funktionen geplant. Eine Java-Debugger soll die Fehlersuche in laufenden Programmen erleichtern und ein sog. Shared Whiteboard ergänzt speziell in der Entwurfsphase von Projekten die textbasierte Kommunikation durch die verteilte Erstellung von Grafiken, Entwurfsskizzen oder komplexeren UML-Diagrammen. Letzteres war ein häufiger Wunsch der Benutzer. Auch die Entwicklung eines VitaminL- Pluggins für Eclipse wäre eine interessante Variante. Hier könnten die Vorteile der mächtigen Entwicklungsumgebung Eclipse mit der gezielten Unterstützung virtueller Teams unter Berücksichtigung ihrer personellen Zusammensetzung kombiniert werden. Ebenso denkbar wäre ein wissensbasiertes System, welches aus einer Menge von potentiellen Teammitgliedern für eine Aufgabe optimale Teams zusammenstellt. Im Rahmen der Erweiterung und Optimierung der Analyse- und Simulationskomponente bieten sich daher noch vielfältige Forschungs- und Entwicklungstätigkeiten an [cf. Kölle et al. 2005].

Nach den beiden Testphasen wird das VitaminL-System im Sommersemester 2006 in den Regelbetrieb übergehen und durch begleitende Benutzertests stetig evaluiert, vor allem im Hinblick auf erweiterte Funktionen. Für dieses Sommersemester ist erneut eine hochschulübergreifende Kooperation mit der Universität Chemnitz in Vorbereitung. Dabei wird der Content per Videokonferenz und VitaminL für die praktischen Übungen exportiert. Interessant wäre hier zu prüfen, ob sich die Einbindung des innerhalb des ELAN-Netzes an vielen Standorten verwendeten Lernmanagementsystem Stud.IP für die Verwaltung der Lehrmaterialien und Teilnehmer anbietet.

Bezüglich der Ergänzung um softwareergonomische Inhalte wird derzeit in einem angegliederten Promotionsvorhaben an der Erstellung eines HCI-Portals gearbeitet, welches mittels des Autorentools Lecturnity² multimedial aufbereitete Lehrmaterialien zur Vorlesung Mensch Maschine Interaktion, virtuelle Tutorien des SELIM-Moduls und ein interaktives Usability Labor miteinander verbindet. Auch hier bildet das Einbeziehen lerntheoretischer Erkenntnisse und die Ausrichtung an konstruktivistischen, problemorientierten Lernprozessen den zentralen Ansatz. [cf. Weichert 2006]

Darüber hinaus wäre eine Erweiterung der Module um Multilingualität, welche aus Kapazitätsgründen im Rahmen des laufenden Projekts nicht möglich gewesen war, insbesondere in Hinsicht auf einen kulturübergreifenden Einsatz erstrebenswert. Dafür wäre jedoch die Frage nach einer Weiterfinanzierung des Projekts zu klären.

² Lecturnity ist ein Produkt der imc AG, welches die Aufzeichnung und Veröffentlichung von Powerpoint-Präsentationen mit den zugehörigen Video- und Tonannotationen ermöglicht.

Literatur

- BISCHOFF, K. (2005): Objektorientierte Softwareentwicklung in virtuellen Teams - Modellierung und Ansätze zur automatischen Erkennung von Problemsituationen. Unveröffentlichte Magisterarbeit im Fachbereich Informations- und Kommunikationswissenschaften der Universität Hildesheim.
- ELAN NIEDERSACHSEN: <http://www.elan-niedersachsen.de>, verifiziert 27.02.2006
- GÖLDNER, A. (2005): Computerunterstützte Kommunikation in virtuellen Teams - Klassifikations- und Lösungsansätze für Problemsituationen in der Chatkommunikation im Rahmen objektorientierter Programmierung. Unveröffentlichte Magisterarbeit im Fachbereich Informations- und Kommunikationswissenschaften der Universität Hildesheim.
- GRÄSEL, C. (1997): Problemorientiertes Lernen. Strategie, Anwendung und Gestaltungsmöglichkeiten. Göttingen: Hofgrefe.
- HENZE, N.; SCHMIDT, C.; WOLPERS, M. (2001): Mediengestützte Didaktik für qualitative Methoden in der Sozialforschung auf der Basis semantischer Modellierung. In: Wagner, E.; Kindt, M. (Hrsg.): Virtueller Campus. Szenarien, Strategien, Studium., Münster et al.: Waxmann, S.164 – 171.
- IABG (2004): Das V-Modell - Planung und Durchführung von IT-Vorhaben, 2004. <http://www.v-modell.iabg.de>, verifiziert 05.03.2004
- KLIMSA, P. (1993): Neue Medien und Weiterbildung: Anwendung und Nutzung in Lernprozessen der Weiterbildung. Weinheim: Dt. Studien-Verlag.
- KÖLLE, R.; LANGEMEIER, G. (2004): Analyse und Unterstützung virtueller Lernteams bei der objektorientierten Softwareentwicklung. In: Bekavac, Bernard; Herget, Josef; Rittberger, Mark (Hrsg.): Information zwischen Kultur und Marktwissenschaft; Proceedings des 9. Internationalen Symposiums für Informationswissenschaft (ISI 2004). Konstanz: UVK. (Schriften zur Informationswissenschaft; Bd. 42), S. 1-22.
- KÖLLE, R.; LANGEMEIER, G.; SEMAR, W. (2005): Programmieren lernen in virtuellen Teams. In: Proceedings des 4. Hildesheimer Evaluierungs- und Retrieval (HIER) Workshop 2005. to appear.
- MANDL, H., GRUBER, H., RENKL, A. (1997): Situiertes Lernen in multimedialen Lernumgebungen. In: Issing, L. J.; Klimsa, P. (Hrsg.): Information und Lernen mit Multimedia. 2. Auflage. Weinheim: Beltz, S.167-178.
- MARGERISON, C. (1990): Team-Management. London: Management Books 2000 Ltd.
- MCMANUS, M. M.; AIKEN, R. M. (1995): Monitoring computer-based problem solving. In: Journal of Artificial Intelligence in Education (1995), Nr. 6, S. 307-336.
- NIEDERSÄCHSISCHES MINISTERIUM FÜR WISSENSCHAFT UND KULTUR: ELAN-Ausschreibung (2. Förderstufe). ELAN-Partner. Mit der Durchführung der Ausschreibung beauftragt ist der strategische Beraterkreis Multimedia. <http://www.elan-niedersachsen.de/fileadmin/Dokumente/Partner-2002-12-17.pdf>, verifiziert 25.02.2006
- RECHENBERG, P.; POMBERGER, G. (Hrsg.) (1999): Informatik-Handbuch. 2., aktualisierte und erweiterte Auflage. München: Hanser.

- SBMM NIEDERSACHSEN: ELAN-Konzept. <http://www.sbmm-niedersachsen.de/ELAN.pdf>,
verifiziert 27.02.2006
- SCHUDNAGIS, M., WOMSER-HACKER, CH. (2002): SELIM: Softwareergonomie für Lernsysteme mit verschiedenartiger lerntheoretischer Basis. In: HAMMWÖHNER, R., WOLFF, CH., WOMSER-HACKER, CH. (Hrsg.) (2002): Information und Mobilität. Optimierung und Vermeidung von Mobilität durch Information. Proceedings des 8. Internationalen Symposiums für Informationswissenschaft.: 7.-10. Oktober 2002, Regensburg, Konstanz: UVK, S. 387f.
- SCHULMEISTER, R. (1997): Grundlagen hypermedialer Lernsysteme: Theorie – Didaktik – Design. 2.Auflage, München: Oldenburg.
- SPENCER, J.; PRUSS, A. (1995): Top Teams - Der Königsweg zu mehr Flexibilität, Effizienz und Erfolg im Betrieb. München: Knaur.
- TMS-ZENTRUM (2003): TMS - das Team Management System. <http://www.tms-online.de>,
verifiziert 20.03.2003
- WEICHERT, STEFFEN (2006): Gestaltung eines multimodalen virtuellen Usability Labors. Unveröffentlichtes Exposé zur Dissertation vom 01.06.2006 im Fachbereich Angewandte Informationswissenschaft der Universität Hildesheim, wissenschaftliche Betreuung durch Frau Prof. Dr. Christa Womser-Hacker.
- WOMSER-HACKER, C. (2004): Interview mit ELAN.
<http://portal.l3s.uni-hannover.de/index.php?id=560>, verifiziert 27.02.2006