

Schlussbericht Projekt Monist

Zuwendungsempfänger : Prof. Dr. M. Egelhaaf, LS Neurobiologie, Fakultät für Biologie, Universität
Bielefeld

Förderkennzeichen: 08NM055a

Vorhabenbezeichnung: MONIST

Laufzeit des Vorhabens: 01.01.2001 bis 30.04.2004

Ablieferung dieses Berichts and die TIB: 30. April 2004

(Ort, Datum)

(Unterschrift, Stempel)

I. KURZDARSTELLUNG

1. Aufgabenstellung

Das Gehirn ist ein überaus komplexes und dynamisches System. Es zu verstehen, gehört zu den großen verbleibenden Herausforderungen der Wissenschaft. In der Forschung helfen Modellsimulationen, das Gehirn zu verstehen. Für das Lernen mit Simulationen im Studium (oder in der Schule) fehlen jedoch klare Einsatzkonzepte: Weder existieren anwendbare Werkzeuge, noch genügend Inhalte, noch eine ausgereifte Didaktik für Modellsimulationen in der Lehre. Das Monist-Projekt hat sich daher der Aufgabe verschrieben, Simulationen in die Lehre der Neuro- und Kognitionswissenschaften zu integrieren.

Kernstück der Entwicklung ist eine Plattform für Simulationen, die eine kritische Masse an Lehrsimulationen und Kursen beinhaltet. Die ‚Monist-Console‘, eine Software, die Zugang zu den Inhalten auf einem zentralen Server erlaubt, bietet verschiedene Arbeitsumgebungen für Lernende, Lehrende und Autoren und berücksichtigt Richtlinien für das Lernen, den Lehreinsatz und die Entwicklung von Modellsimulationen.

Das Bielefelder Teilprojekt hatte drei Aufgabenbereiche.

- (a) Inhaltsproduktion und Einsatz: Im Rahmen von ‚Autorenarbeitspaketen‘ wurden Simulationen an den verschiedenen Standorten des Verbundes entwickelt und getestet. In Bielefeld wurden drei Autorenarbeitspakete abgewickelt, nämlich "Sensorik/Neurobiologie" (Egelhaaf), "Motorik/Theoretische Biologie"(Cruse) und "Neuronale Netze" (Ritter) .
- (b) Softwareentwicklung*: Die Plattform für Simulationen (Monist-Server und Monist-Console) wurde komplett in Bielefeld entwickelt.
- (c) Koordination / Redaktion: als federführender Standort im Verbund hatte Bielefeld die Aufgabe, die Kooperation zu gestalten und zu begleiten.

Die drei Teilbereiche (a) Inhalte, (b) Software und (c) Koordination werden im Folgenden jeweils nacheinander behandelt.

* Begriffsbestimmungen: Wenn von ‚Inhalten‘ die Rede ist, sind meist ‚Simulationen‘ gemeint. Simulationen sind selbstverständlich Software. Wenn aber von ‚Software-Entwicklung‘ die Rede ist, sind ausschließlich die Arbeiten für die zentrale Simulations-Plattform in Bielefeld gemeint, auch wenn für die Produktion der Inhalte Software-Entwicklung notwendig ist. Inhaltsproduktion ist Bestandteil der Autorenarbeitspakete.

2. Voraussetzungen

Die Situation, die zu der Projektinitiative geführt hat, lässt sich treffend als „Lücke zwischen Forschung und Lehre“ charakterisieren: In der neuro- und kognitionswissenschaftlichen Forschung wird massiv mit Modellsimulationen gearbeitet – in der Lehre hingegen spielen Modellsimulationen keine oder eine untergeordnete Rolle. Die Funktion von Modellsimulationen in der Forschung kann sehr gut als „Lernwerkzeug“ beschrieben werden: Simulationen helfen, die scheinbar undurchdringliche Komplexität und Dynamik des Gehirns zu entschlüsseln. Wenn Simulationen Lernwerkzeuge sind, warum werden sie dann so wenig in der Lehre eingesetzt? Hierfür gibt es gute Gründe: Simulationen sind anspruchsvolle Medien für alle Beteiligten, Autoren, Lehrende und Lernende. Dies wird voraussichtlich auch so bleiben, denn, überspitzt formuliert, gilt: „Komplizierte Inhalte – komplizierte Lehre!“.

Das heißt, wenn man komplizierte Inhalte vermitteln möchte, muss man auch damit rechnen, dass mehr Investitionen als bei der Vermittlung einfacher Inhalte geleistet werden müssen. Simulationen vereinfachen zwar das Verständnis von komplexen und dynamischen Systemen – die Systeme an sich (und damit die Lerninhalte) können jedoch nicht beliebig weit vereinfacht werden, weil ihnen dann die Komplexität und Dynamik verloren ginge, die es eigentlich zu lernen galt.

In den beteiligten Bielefelder Einrichtungen (Cruse, Egelhaaf, Ritter) gab es vor Projektbeginn schon umfassende und innovative Ansätze zum Einsatz von Simulationen in der Forschung und auch in der Lehre. Trotzdem verursachte die zuvor beschriebene allgemeine Problematik, bestimmte Bedarfe, die sich im Bielefelder Teilprojekt in verschiedenen Aspekten und verschiedenen Ausprägungen zeigten:

(a) Inhalte

Zahlreiche Modelle zu biologischen, psychologischen und informatischen Themen lassen sich in der grundständigen Lehre besser und effektiver mit Simulationen verstehen als mit herkömmlichen Lehrmedien. Da die Entwicklung von Simulationen aufwändig ist, kann sie mit den Mitteln der Grundausstattung nur in eingeschränkter Masse erfolgen. Daher war in den Bielefelder Einrichtungen ein Mangel an Inhalten zu beobachten. Es gab viele wissenschaftliche Simulationen, die sich jedoch nur sehr eingeschränkt für die Lehre eignen – ihnen fehlte häufig die Anleitung und die konkrete Aufgabenstellung und überforderte die meisten Lernenden. D.h. zusätzlich war ein mangelnde didaktische Eignung der Inhalte zu beobachten. Darüber hinaus gab es nicht ausreichend Richtlinien, wie Lehrende, die Simulationen in ihre Lehre einbinden können, selbst wenn sie eine gute Simulation für ein Modell zur Verfügung haben: es fehlten intelligente und erprobte Einsatzkonzepte! Dies betrifft im Allgemeinen nicht nur didaktische, sondern auch technische und infrastrukturelle Gegebenheiten. Alltägliche Beispiele sind die fehlenden Instrumente zur Lernerfolgskontrolle, ein falsches Betriebssystem auf dem Präsentationsrechner oder mangelnde Rechnerplätze, um das Selbstlernen zu ermöglichen. Diese Probleme bringen auch Schwierigkeiten mit sich, die Simulationen an anderen Standorten und in anderen Einrichtungen einzusetzen.

(b) Software

Von technischer Seite ist vor allem zu bemerken, dass es kaum Standards für Simulationen gab. Es existierten zahlreiche Ansätze zur Entwicklung von Simulationen (meist auf Programmiersprachen wie C oder JAVA basierend), was zu einer schwierig zu kontrollierenden technischen Heterogenität in der Masse der einzelnen Simulationen führen konnte. Darüber hinaus gab es keine Standards für Dokumentation, geschweige denn Rezepte für die didaktische und inhaltliche Gestaltung. Dies alles führt im Allgemeinen zu der Situation, dass im Bildungssystem (national und international) zwar ausreichend Simulationen existieren (z.T. auch im Internet abrufbar), aber der institutionsübergreifende Einsatz nur sehr schwer möglich ist. Oftmals wird die Simulation an jeder Einrichtung erneut implementiert. Beispiel: Die Simulation „Travelling Salesman“ findet sich allein auf dem Internet in mehreren hundert Varianten mit häufig wiederkehrenden Algorithmen. Die technische Heterogenität zeigte sich auch an den beteiligten Bielefelder Einrichtungen. Zur Entwicklung didaktischer Standards wurden bereits vor Monist andere Projekte durchgeführt (z.B. RUBIN, gefördert durch das Ministerium für Schule und Wissenschaft NRW).

(c) Koordination

Die inhaltlichen und technischen Probleme hatten ihre Entsprechungen im organisatorischen Bereich. Da es nicht genügend Standards und nicht genügend passende Inhalte gab, kam jede

Bildungsinstitution, die mit Simulationen lehren wollte, immer wieder in die Situation, Inhalte neu produzieren zu müssen. Dies kann aber mit den im Forschungs- und Lehralltag zur Verfügung stehenden Ressourcen nur punktuell gelingen. Auch dann ist meist noch viel „Expertenwissen“ in den Einsatz der Simulation integriert (etwa konkrete Einsatzszenarien, mündliche Einleitungen und vom Lehrenden individuell spezifizierte Erfolgskontrollen), die den Einsatz einer Simulation in anderen Standorten oft scheitern lassen. Es fehlte also an koordinierenden Strukturen, die sowohl eine Übertragbarkeit durch die Einhaltung von Standards ermöglichen als auch eine thematische Aufteilung erlauben, die Doppelproduktionen vermeiden hilft. Darüber hinaus gibt es im Allgemeinen häufig genug rechtliche Fragen (z.B. Welche Lizenz gilt für die Benutzung?) oder eine allgemeine Zurückhaltung, weil vermutet wird, dass die eigenen intellektuelle Leistungen unkoordiniert weitergegeben werden oder unter anderem Namen veröffentlicht werden. Kurzum: Es fehlte an einer Veröffentlichungsstrategie, die den durchaus verbreiteten Ethos von Forschenden und Lehrenden (in ihrer Funktion als Autoren) unterstützt, ihre Arbeiten zu teilen und gemeinsam zu nutzen.

3. Planung und Ablauf des Vorhabens

Das unausgeschöpfte Potential von Simulationen in der Lehre der Neuro- und Kognitionswissenschaften, das mit den technischen, inhaltlichen und koordinatorischen Problemen (s.o.) zu erklären ist, führte in Bielefeld bereits vor dem Monist-Projekt zu Initiativen (z.B. RUBIN) und letztendlich auch zu der Idee, ein bundesweites Netzwerk für Simulationen zum Thema Gehirn aufzubauen. Die grundsätzliche Architektur des Netzwerkes sah eine Aufteilung in „Expertenstandorte“ (Autorenarbeitsplätze) vor, d.h. acht Themen- und Methodenschwerpunkte, sowie einen zentralen Evaluationsstandort in Freiburg, eine zentrale Software-Entwicklung in Bielefeld und die Federführung („Koordination und Redaktion“) in Bielefeld.

(a) Inhalte

Die Themen- und Methodenschwerpunkte wurden auf verschiedene Standorte in Deutschland verteilt, die sich durch spezifische Expertise auszeichneten. Mit unterschiedlich starker Ausprägung wurden Simulationen bereits vor dem Projektbeginn zu Lehr- und Forschungszwecken eingesetzt. In Bielefeld wurden die drei Autorenarbeitspakete "Sensorik/Neurobiologie" (Prof. Dr. Egelhaaf, Lehrstuhl für Neurobiologie, Fakultät für Biologie), "Motorik/Theoretische Biologie" (Prof. Dr. Cruse, Abteilung für biologische Kybernetik und Theoretische Biologie, Fakultät für Biologie) und "Neuronale Netze" (Prof. Dr. Ritter, AG Neuroinformatik, Technische Fakultät) abgewickelt. Nach einer verbundweiten Bestandsaufnahme und Bedarfsanalyse wurden die zu entwickelnden Simulationen spezifiziert (jeweils ein Autorenarbeitspaket pro Einrichtung). Entgegen der ursprünglichen Planung, die einen stufenförmigen Entwicklungsprozess der Simulationen („Wasserfall“) vorsah, wurde auf den Ergebnissen der Bedarfsanalyse basierend durch die Verbundversammlung auf dem ersten Treffen beschlossen, dass die Lehreinheiten zunächst auf den konkreten Bedarf der Einrichtungen über die gesamte Projektlaufzeit in zyklisch-kontinuierlicher Weise zu produzieren („Evolution“) sind und zu Ende der Projektlaufzeit für den standortübergreifenden Einsatz in die zentrale Plattform (s.u. Software-Entwicklung) integriert werden. Da somit die Bedarfe der Einrichtungen die Plattform bestimmten und nicht umgekehrt, wurde die Plattform ebenfalls evolutionär entwickelt. Diese Form der Entwicklung stellt die Eignung der Simulationen für die Lehre sicher, da sie nahtloser in Lehrveranstaltungen eingesetzt und – auf den Evaluationsergebnissen aufbauend – optimiert werden können. Das interdisziplinäre Curriculum für Simulationen zum Thema Gehirn bekam

nicht die globale Struktur einer ‚Schule‘ (Studiengang, Studienabschnitt) sondern die Struktur eines Baukastens (‚Repository‘). Im Hinblick auf die Qualitätssicherung, die Überführung in den Regelbetrieb und die Modularität (vielseitige Einsatzszenarien) hat sich diese Planung im Ablauf des Vorhabens in Bielefeld bewährt.

(b) Software-Entwicklung

Aufgabe der zentralen Software-Entwicklung in Bielefeld war die Entwicklung einer Plattform, die die Inhalte der Autorenpaketpakete bündeln kann und Richtlinien vorgibt, wie Simulationen zu dokumentieren und einzusetzen sind. Diese Standardisierungsmaßnahmen wirken den oben beschriebenen Problemen der technischen Heterogenität und didaktischen Mangelhaftigkeit entgegen (s.a. I.2). Insbesondere waren hier zwei Entwicklungsstränge von Bedeutung: Die Weiterentwicklung eines für die Lehre geeigneten Formats für Simulationen, das Anleitungen für Lernende enthält (sog. ‚Lehrsimulationen‘, die sich bereits im RUBIN-Projekt bewährt hatten) und die Entwicklung einer Simulations-Plattform (eines Document- bzw. Learning-Management-Systems), die ‚Monist-Console‘. Lehrsimulationen stellen das angewendete didaktische Format dar: durch interaktive Parametervariation und dynamische Visualisierung werden wissenschaftliche Modelle erfahrbar. Genaue Instruktionen und Aufgabenstellungen für Lerner unterstützen einen interaktiven und individuellen Arbeitsprozess mit dem Modell. Hinzu kommt eine inhaltliche und methodische Einbettung der Modelle über Fachtexte mit Zugang zu Glossar, Literaturhinweisen sowie interne und externe Verweise auf andere Simulationsprogramme. Als standardisierte Dialoge eingebettet in die Monist-Console unterstützt dieses ‚Rezept‘ Autoren bei der Erstellung der Simulationen und bietet Lernenden als wiederkehrende Struktur eine Orientierungshilfe beim Umgang mit sonst manchmal unzugänglichen Simulationen. Der zweite wichtige Entwicklungsstrang – die Entwicklung der Monist-Console – sollte das Format der Lehrsimulationen unterstützen, aber auch die Einbettung anderer Simulationstypen erlauben. Insgesamt ergab sich durch diese Anforderungen folgende Spezifikation (Arbeitspakete in Klammern): „Alle Inhalte werden über die Console bereitgestellt. Die Console ist on- und offline voll lauffähig. Nutzer sind dadurch nicht auf einen permanenten Online-Zugang angewiesen. Die Console beinhaltet Schreibtische für die verschiedenen Nutzergruppen (Studienschreibtisch und Lehrschreibtisch, Autorschreibtisch mit Drehbuch-Skripter, Lehreinheiten-Skripter, und Simulationen-Skripter, Autoren-GUI) einen Datenbankzugang sowie die Client/Server-Kommunikation (Online-Umgebung mit Monist-Server, Monist-Media-Datenbank, Internet-Kurssystem). Ein Kommunikationssystem erleichtert die Gruppenkommunikation zwischen Monist-Nutzern und ist auch die Basis für Online-Seminare. Der Studienschreibtisch bietet Zugang zu Kursen und Lehrsimulationen. Eigene Texte, wie Notizen, Protokolle, etc., können im Studienschreibtisch erstellt und verwaltet werden. Das Kommunikationssystem erleichtert den Austausch von Informationen und Material und gewährt einen Zugang zum Monist-Server. Der Lehrschreibtisch bietet neben den Grundmodulen der Monist-Console spezielle Funktionen zur Verwaltung von Kursen. Auch stellt er die Möglichkeit bereit, Kursteilnehmende als Gruppen zu organisieren und bestimmte Zugangsberechtigungen zum System zu erlauben. Über das Kommunikationssystem ist die Organisation eines Online-Seminars möglich. Der Autorschreibtisch enthält Werkzeuge, die für eine Produktion von Lehrsimulationen notwendig sind. Hierzu gehören vor allem ein Lehrsimulationseditor und ein Kurseditor. Der Lehrsimulationseditor besteht aus verschiedenen Modulen, die die Einbettung von Text, Simulationen, Formeln, Aufgaben etc. erlauben. Bestehende Lehrsimulationen können über einen Kurseditor zu neuen Kursen re-/kombiniert werden.“ Die geforderte Funktionalität geht über den Funktionsumfang eines gewöhnlichen Document- oder Learning-Management-Systems hinaus, v.a. was die Unterstützung von Simulationen angeht (d.h. deren Einbindung, Dokumentation und Offline-Verfügbarkeit). Tatsächlich ergab die Bestandsaufnahme und Analyse von kommerziellen und frei verfügbaren Systemen, dass

kein bestehendes System diese Anforderungen erfüllt. Darüber hinaus würde das Projekt mit einer Lizenzabhängigkeit ‚infiziert‘, die weder von Projektnehmerseite noch von Projektgeberseite gewünscht war. Die Plattform wurde daher als Open-Source Entwicklung unter GPL Bedingungen auf JAVA/XML/MySQL basierend entwickelt.

(c) Koordination

Der Mangel an koordinierter Produktion und Nutzung von Simulationen in verschiedenen Bildungsinstitutionen veranlasste im Jahr 2000 dazu, ein Netzwerk für den Bereich der Neuro- und Kognitionswissenschaften aufzubauen, das sich zur Durchführung des Monist-Projektes in seiner jetzigen Organisation formierte. Das Netzwerk deckt grundlegende Gebiete zum Thema Gehirn durch seine interdisziplinäre Struktur ab, die sowohl Biologie und Psychologie als auch informatische und technisch-angewandte Aspekte berücksichtigt. Es ist als Kristallisationspunkt für eine flächendeckende Behandlung des Themas angelegt. Formierung des Netzwerkes, Antragstellung, Kooperationsvereinbarung, Bestandsaufnahme und Bedarfsanalyse sowie Zielvereinbarung wurden vom federführenden Standort Bielefeld begleitet, ohne die Souveränität der Partner einzuschränken. Bestimmendes Organ war stets die Verbundversammlung (gemäß Kooperationsvereinbarung). Zur Abstimmung der Finanz- und Aufgabenplanung und der Spezifikation der Inhaltsproduktion (Redaktion) wurden neben den üblichen Kommunikationswegen jährlich drei (also insgesamt neun) Verbundtreffen organisiert und ein internetbasiertes Redaktionssystem eingerichtet, das die administrative Abwicklung und Kooperation kontinuierlich unterstützte. Darüber hinaus wurden die einzelnen Standorte besucht. Die Maßgabe der Übertragbarkeit der Entwicklungen wurde über die Abstimmung auf Treffen, Reisen etc. hinaus durch ein verbundweit geltendes Rechte- und Lizenzmanagements sowie dessen Operationalisierung als Veröffentlichungsstrategie in der Monist-Console unterstützt. Ferner wurden zum Zwecke der Streuung Veröffentlichungen und Projektpräsentationen organisiert. Zum Bereich Koordination gehörte auch die interne Organisation der Bielefelder Entwicklungen, besonders der Software-Entwicklung und die administrative Abwicklung (Finanz- und Aufgabenplanung, Berichtswesen etc.).

4. Wissenschaftliche und technische Voraussetzungen

(a) Inhalte

Gemäß der festgelegten Themen- und Methodenschwerpunkte für die Autorenarbeitspakete war die wissenschaftliche Beurteilung der Simulationen Aufgabe der zuständigen Bielefelder Einrichtungen, da dort jeweilige Expertise vorhanden war und Konzepte sowie Optimierungen innerhalb der Einrichtungen und in den Lehreinheiten erarbeitet werden konnten. Technisch gab es die globale Maßgabe, möglichst keine lizenzpflichtigen Fremdentwicklungen in die Inhalte einzubringen, die den Richtlinien des Verwertungsplanes widersprachen. Zudem sollte möglichst plattformübergreifend für Windows und Unix-basierte Systeme (z.B. Linux), aber auch MacOS entwickelt werden. Ferner sollten keine technischen Ansätze verwendet werden, die Schwierigkeiten bei der Integration in die Monist-Console machen könnten. Für die Einrichtung Egelhaaf wurden – vor allem aufbauend auf den Entwicklungen des RUBIN-Projektes – ausschließlich JAVA-Applikationen entwickelt. Auch für die Einrichtung Cruse wurden JAVA Applikationen entwickelt, aber auch in HTML/CGI Strukturen eingebettete Applets sind zu erwähnen. Für die Einrichtung Ritter wurden C-Programme erarbeitet, die auf Bibliotheken des Simulationswerkzeuges Neo/NST zurückgriffen. Zur Erfüllung der o.g. technischen Richtlinien wurde Neo/NST für Windows portiert und die Monist-Console mit einem spezifischen Lademechanismus für Neo/NST Simulationen versehen.

(b) Software-Entwicklung

Die wissenschaftlichen und technischen Voraussetzungen für die Entwicklung der Monist-Console waren zum Teil gegeben, mussten aber zum Teil auch neu entwickelt werden, weil bestimmte Anforderungen, die die Verwaltung von Simulationen in einem Document-Management-System mit sich bringt, von keinem anderen bekannten System verwirklicht wurden (s.o.). Soweit es möglich war, wurde auf gängige Standards zurückgegriffen – in den innovativen Bereichen mussten Voraussetzungen erst geschaffen werden. Die Architektur des Systems sollte optimale Möglichkeiten zur Erweiterbarkeit, Anschlussfähigkeit und Schnittstellenentwicklung bieten. Dieselben technischen Richtlinien wie bei der Entwicklung der Simulationen wurden auch der Monist-Console zugrunde gelegt, was zu folgenden Überlegungen führte: JAVA als Basis der Applikation ist plattformübergreifend und unterstützt alle gängigen Internet-Protokolle optimal, XML ist der aktuell meist verbreitete Standard in der Datenbeschreibung und MySQL ein Standard in der Datenverwaltung. Das System sollte, wenn möglich, vorhandene Bibliotheken verwenden, die der GPL (GNU General Public License) unterliegen und wurde somit als typisches Open-Source Projekt angelegt. Bei der Datenhaltung (Metadaten) wurde zur Wahrung der Kompatibilität mit internationalen Standards von Sacherschließungsinstrumenten im Bibliothekswesen die Dublin-Core Spezifikation berücksichtigt. Für die lernspezifischen Metadaten wurden Elemente des IMS (IEEE-LOM) verwendet. Das Format der Lehrsimulation, das in die Benutzerfläche der Monist-Console integriert wurde, enthält Elemente des „Instructional Design“, einem pädagogischen Standard zur Gestaltung von Lerninhalten.

(c) Koordination

Die wissenschaftlichen Voraussetzungen waren in den beteiligten Einrichtungen gegeben und mussten von der Koordination nicht unterstützt werden. Für die koordinierte Nutzung und Entwicklung technischer Voraussetzungen wurden die o.g. Richtlinien spezifiziert und deren Umsetzung begleitet.

5. Zusammenarbeit mit anderen Stellen

Die Zusammenarbeit mit den Teilprojekten an den anderen deutschen Standorten fokussierte größtenteils auf Inhaltsproduktion, d.h. auf die Abwicklung der Autorenarbeitspakete. Das Teilprojekt Freiburg, Seel hatte die spezifische Aufgabe, die begleitende Evaluation durchzuführen und arbeitete somit zusammen mit den Bielefelder Lehrenden, die die Monist-Entwicklungen einsetzten.

Darüber hinaus gab es eine kontinuierliche Abstimmung zwischen allen Teilprojekten über Verbundversammlung, Standort-Besuche und sonstige Kommunikationsformen. Wissenschaftsgesellschaften wurden durch Präsenz an Tagungen einbezogen (z.B. Deutsche Neurowissenschaftliche Gesellschaft, European Cognitive Science Society). Die Veranstaltungsangebote des Projektträgers „Neue Medien in der Bildung“ (DLR) zu Rechtsfragen, eLearning-Standards, Verwertung etc. wurden z.T. mit Referentenbeteiligung wahrgenommen. Mit dem Projektträger wurde ferner in administrativen Fragen, aber auch in verschiedenen Sachfragen zusammengearbeitet, z.B. Im Rahmen der Erstellung des „Kursbuches eLearning 2004“ oder der Mitarbeit in der Referenzgruppe Nachhaltigkeit.

II. ERGEBNISSE

1. Übersicht der Ergebnisse

Der größte Teil der Ergebnisse ist Software, nämlich die Lehrmedien (Simulationen) und die Simulations-Plattform (Monist-Console und Monist-Server). NutzerInnen können sich auf der Internetseite <http://www.monist.de> anmelden und die Monist-Console herunterladen, wenn sie ihre Registrierungsdaten erhalten haben. Nach Installation der Console, können sie sich mit dem Monist-Server verbinden und die Simulationen herunterladen und benutzen.

Im Einzelnen sind folgende Ergebnisse zu nennen (vgl. I.3):

(a) Inhalte

Insgesamt stehen aus Bielefeld nach Projektende ca. 80 Simulationen zur Verfügung (siehe Zwischenberichte 2001-2003), die zu ähnlich großen Anteilen aus den Autorenarbeitspaketen I-III) resultierten. Die Tabelle „Inhalte“ im Anhang 1 gibt eine Übersicht. Um den Umfang der Ergebnisse einschätzen zu können, ist es wichtig die Bandbreite der Entwicklungen zu berücksichtigen: Es gibt unterschiedliche Einsatzszenarien – von Präsentation in Vorlesung und Seminar bis zum Selbstlernen in Praktikum oder Übung oder daheim im Online-Kurs. Und es gibt unterschiedliche Einsatzdauern (von 5 Minuten bis zu 5 Tagen). Um trotz dieser Bandbreite einen Eindruck zu vermitteln, welchen Umfang eine Simulation hat, sei ein Einsatzszenario beschrieben, das sich in Bielefeld häufig wiederholte: Diese typische Simulation wird in Präsenz-Veranstaltung in Rechnerräumen (oder Seminarräumen mit Notebooks) eingesetzt (1-2 Lerner pro Rechner) und hat eine Bearbeitungsdauer von 1-2 Stunden (Einsatzdauer 2-4 Stunden). Es gibt Vor- und Nachbesprechung bzw. Ergebnisbericht und die Möglichkeit Zwischenfragen zu stellen („Blended Learning“). Als globale Erfolgskontrolle ist ein Protokoll anzufertigen, das von den Lehrenden korrigiert wird.

Trotz der Bemessungsschwierigkeiten, die die Bandbreite der Simulationen mit sich bringt, sei zu Übersichtszwecken im Folgenden eine Schätzung der Gesamteinsätze in Bielefeld gegeben: Fast jede der ca. 80 Simulationen wurde ein- bis mehrmals eingesetzt (insgesamt ca. 500 Einzeleinsätze). Sie wurden insgesamt von ca. 15 Lehrenden in ca. 20 verschiedenen Veranstaltungen getestet und unterstützten dabei ca. 1000 verschiedene Lernende, v.a. in den Studiengängen Biologie (Diplom), Biologie (Lehramt), Gesundheitswissenschaften (Diplom), Naturwissenschaftliche Informatik (Diplom), sowie neuerdings diverse BA- und MA-Studiengänge. Allein mit den in Bielefeld produzierten Inhalten ließe sich – in der Summe der Bearbeitungszeit – ein größerer Studienabschnitt oder ein kurzer Studiengang (BA oder MA) füllen. (Diese Angabe dient nur der Veranschaulichung und beinhaltet keine Empfehlung für die Gestaltung eines Online-Studiengangs – im Gegenteil: Es wurde stets ein Anreicherungskonzept von Präsenzstudiengängen verfolgt und nicht der Ersatz bestehender Lehre.)

Die Simulationen können nach Installation der Monist-Console (s.u. bzw. <http://www.monist.de>) vom Server heruntergeladen werden. Sie liegen größtenteils als Lehrsimulationen vor, d.h. sie sind dokumentiert mit Fachtexten, Anleitungen und Aufgaben (Englisch, Deutsch oder zweisprachig). Metadaten zu thematischen, didaktischen und technischen Voraussetzungen erlauben einen schnellen Überblick. Mehrere Simulationen können zu Kursen zusammengefasst werden. Als modularer „Baukasten“ erlaubt dieser Ansatz Lehrenden die Gestaltung von Einzeleinsätzen oder Kursen, wie sie an ihrer Einrichtung gefordert sind.

(b) Software-Entwicklung

Die Simulations-Plattform besteht aus dem Monist-Server und der Monist-Console. Der Monist-Server ist an der Universität Bielefeld lokalisiert und hält die Lehrmedien (Simulationen) zum Download sowie andere zentrale Daten (z.B. Registrierungsdaten) und diverse Dienste (Aufgabenverwaltung für Lehrende) bereit. Der Funktionsumfang der Console hängt von den Rechten der Nutzer ab, wie im Folgenden beschrieben.

„Lernende“ haben alle Grund-Funktionen, die sie benötigen, um Simulationen vom Server zu laden, zu starten und zu benutzen.

-- Der Navigator ermöglicht über eine Baumstruktur den Zugriff auf alle Inhalte, Kurse und Benutzer des Monist-Systems. Die aktuellen Lehrsimulationen können mit dem Content Browser angesehen werden.

-- Lernmanagement: Aufgabenstellungen sind in Lehrsimulationen integriert. Studenten können ihre Lösungen auf den Server laden und Lehrende haben Zugriff auf deren Lösungen. Komplette Protokolle werden durch Formulare und Lehreinheiten unterstützt.

-- Das Kommunikationssystem unterstützt synchrone (chat) sowie asynchrone Kommunikation (eMail, News) zwischen Kursteilnehmern und Tutoren innerhalb des Systems. Auf diese Art und Weise können Probleme, Lösungen von Aufgaben oder Korrekturen diskutiert werden.

„Lehrende“: Lehrsimulationen können in beinahe jedem Lernszenario in der Hochschullehre eingesetzt werden, z.B. in Tutorien und Vorlesungen. Wegen seiner Offline- und Online-Fähigkeiten unterstützt die Monist Console Verzahnung mit der Präsenzlehre ebenso wie reines Online-Lernen. Obwohl die Console das Selbststudium unterstützt, hilft es Dozenten insbesondere, Simulationen in ihre Kurse zu integrieren. Mit seinem Online-File-Management und dem Kommunikations-System hilft das Monist-System eigene Kurse zu erstellen. Das Kommunikations-System begünstigt ebenso Lernen in Gruppen auch über größere Distanzen hinweg. Das System ist nicht auf das Lernen mit Modellsimulationen beschränkt. Prinzipiell unterstützt Monist jede Form von Lernmaterial. Zusätzlich können 'klassische' Lehrinhalte mit Lehrsimulationen kombiniert werden.

„Autoren“: Um einen einfachen Produktionsprozess für Lehrsimulationen auch für Nicht-Programmierer zu ermöglichen, sieht die Console einen Autorenmodus vor, mit dem Lehrmaterial erstellt und verändert werden kann. Schon existierende Simulationen (Applets, Programme, etc.) können sehr leicht eingebunden werden. Lehrmaterial wird durch ein standardisiertes Metadaten-Set gekennzeichnet, um die Wiederverwendbarkeit des Materials durch verschiedene Nutzer zu maximieren. Die Console bietet die Möglichkeit, Inhalte zu veröffentlichen. Sobald eine Lehreinheit auf den Server geladen und vom Autor für die Öffentlichkeit (oder eine Gruppe) freigegeben wird, erhält die inhaltliche Redaktion content@monist.de eine Benachrichtigung. Nach Sichtung durch die Redaktion wird die Einheit dann veröffentlicht oder für eine Evaluierung (Beta-Test) vorgesehen. Es besteht für Autoren alternativ die Möglichkeit, direkt einen Beta-Test anzufordern. Inhalte können sowohl für einzelne Gruppen (z.B. bestimmte Kurse) oder Einzelpersonen freigeschaltet werden.

„Entwickler“: Das Client-Server-System basiert auf JAVA, XML und MySql. Die Console ist ein Open-Source Projekt (GPL-Lizenz). Das heisst, Entwickler können an der Programmoptimierung und Weiterentwicklung teilhaben. Das Kommunikationssystem der Monist-Console beinhaltet Möglichkeiten, Ideen einzubringen und Fehler zu bemerken. Die Entwicklung wird moderiert durch die technische Redaktion developer@monist.de.

(c) Koordination

Die organisatorischen und rechtlichen Rahmenbedingungen des Projekts wurden durch die Kooperationsvereinbarung und Lizenzvereinbarungen für Software und Simulationen spezifiziert. Die Software unterliegt der GPL-Lizenz (s.o.), deren Zustimmung Bedingung für die Installation der Software ist. Die Inhalte werden durch die Monist-Content-License (MCL) bestimmt, deren Zustimmung auch Bedingung für die Installation ist. Das thematische Netzwerk (der Verbund) ist durch die Kooperationsvereinbarung eine fortbestehende Organisationsform (MAG: fortbestehende Monits-Arbeits-Gruppe). Zur Abwicklung der wichtigsten Arbeitsbereiche wurde durch die Verbundversammlung ein inhaltlicher, ein technischer und ein inhaltlicher Redakteur festgelegt. Die Redakteure stimmen anfallende Aufgaben mit den betroffenen Personen im Netzwerk ab. Soweit möglich wurden administrative Aufgaben an Dauerstelleninhaber (Professoren, festes wissenschaftliches Personal wie akademische Räte und nicht-wissenschaftliches Personal, wie EDV AdministratorInnen) mit entsprechenden Anweisungen und Schulungen übertragen. Eine Projektdokumentation wird erstellt und voraussichtlich im Eigenverlag veröffentlicht (ISBN 3-00-013629-0).

2. Voraussichtlicher Nutzen

Das Monist-Projekt ist durch entsprechende Maßnahmen (s. II.1) auf den Dauerbetrieb ausgelegt. Der unmittelbare Nutzen für die Lernenden an den beteiligten Institutionen ist offenkundig: Sie können in ihrer Ausbildung fortwährend von der verbesserten Lehre profitieren. Das Angebot steht Lernenden, Lehrenden und Autoren anderer Institutionen offen und kann somit auch dort den entsprechenden Nutzen erbringen. Es ist ferner vorgesehen, die Projektergebnisse zu verwerten, um erweiterte oder neue Ergebnisse zu erzielen. Hierbei wird zwischen den Verwertungsformen Betrieb, Pflege, Optimierung, Weiterentwicklung und Neuentwicklung unterschieden. Die im Bereich Inhalte und Software erweiterbare Anlage der Projektergebnisse (s.a. II.1) wird durch die Monist-Arbeits-Gruppe (MAG), insbesondere durch die Redaktionen koordinierend unterstützt. Erstere Verwertungsformen können durch die Institutionalisierung und Einbindung externer Leistungen realisiert werden, während letztere Verwertungsformen zusätzlicher Einnahmequellen bedürfen. Eine genaue Beschreibung dieses Verwertungsplans wurde in dem Konzeptpapier „Monist 2004 und danach“ aus dem November 2003 von (Wolfram Horstmann, Sören Lorenz und Martin Egelhaaf) ausgeführt (Anlage 2). Weitergehender Nutzen der Projektergebnisse ließe sich erzielen, wenn die Ergebnisse generalisiert würden, z.B. indem die Monist-Console als generisches Learning-Management-System (zum Thema Gehirn) ausgebaut würde oder die Simulationen nicht auf Thema Gehirn beschränkt blieben, sondern themenübergreifend eingesetzt würden. Für letztere Perspektive wurde das Szenario „Simulationen als Bildungsmedien“ (Wolfram Horstmann, Sören Lorenz und Martin Egelhaaf) im Dezember 2003 (Anlage 3) entwickelt.

3. Fortschritte an anderer Stelle

Nach wie vor ist kein mit der Monist-Console vergleichbares Learning- oder Document-Management-System bekannt, das Simulationen in dieser Form verwalten und aufrufen kann. Trotzdem entwickelt sich das Potential der zahlreichen Systeme extrem schnell und bringt möglicherweise nach einer Konsolidierungsphase auch durchschlagende technische Standards oder Standard-Systeme hervor. Das Monist-Projekt erfüllt alle Voraussetzungen für die Berücksichtigung neuer Standards oder die Schaffung von Schnittstellen zu anderen Systemen.

(Es war weder Anspruch noch Wunsch des Monist-Projektes ein LMS zu entwickeln; es war aber notwendig, um die dem Konzept entsprechende Funktionalität zu erreichen.) Sollte sich also ein System (oder mehrere Systeme) durchsetzen, sind Maßnahmen zum Zwecke der Integration von Funktionalität möglich und erwünscht. Im inhaltlichen Sektor werden Fortschritte an anderer Stelle ebenfalls erzielt. Weltweit werden laufend Simulationen zum Thema Gehirn produziert. Es ist nicht auszuschließen, dass ähnliche Simulationen ebenfalls an anderer Stelle entwickelt wurden. Der Mangel an Dokumentation und Anleitung, der zu Schwierigkeiten im Lehreinsatz und die Übertragbarkeit einschränkt (s.a. I.1.) ist jedoch weiterhin häufig zu beobachten. Eine ähnliche Initiative, die das Ziel hat diesen Mangel zu beheben, ist bis heute nicht bekannt geworden. Es ist ein mittelfristiges Ziel des Monist-Projektes externe Simulationen zum Thema Gehirn in die Plattform zu integrieren.

4. Veröffentlichungen

(a) Print (Auswahl)

- Simulationen zum Thema Gehirn (Projektdokumentation ‘Monist’) ISBN 3-00-013629-0 (in Vorbereitung)
- Explaining Brains by Simulation (Horstmann W.) Doctoral Dissertation, Universität Bielefeld, 272 Seiten (2004)
- Lernen mit Simulationen zum Thema Gehirn: Richtlinien für Software, Interaktionsdesign und Einsatz – Arbeitstitel – (Lorenz S.) Dissertation, Universität Bielefeld (in Vorbereitung)
- Ein Simulation-Management-System: Softwarearchitektur und Usability – Arbeitstitel – (Oesker M.) Dissertation, Universität Bielefeld (in Vorbereitung)
- MONIST – Simulationen zum Thema Gehirn. in „Kursbuch eLearning 2004“ Herausgeber: Deutsches Zentrum für Luft- und Raumfahrt - Projektträger „Neue Medien in der Bildung + Fachinformation“ im Auftrag des Bundesministeriums für Bildung und Forschung (2004)
- The monist-project: educational simulations for brains (Horstmann W., Lorenz S., Egelhaaf M.) In: Proceedings of the 29th Göttingen Neurobiology Conference 2003 (Elsner N., Zimmermann H., eds), 1064. Thieme: Stuttgart. (2003)
- Standardizing simulations: Uphill all the way (Horstmann, W.) In "Campus 2002" eds. Bachmann, G., Haefeli, O., & Kindt, M., pp. 218-230. Waxmann, Zürich. (2002) 3-00-013629-0

(b) Präsentationen (Auswahl)

- ED-Media - Internationale Konferenz für Bildungsmedien. Präsentation, Buchbeitrag zu MONIST. Lugano, Schweiz (2004)
- LearnTec - Internationale Messe und Kongress für Bildungstechnologie. Vortrag und Messe-Präsentation von MONIST auf dem Messestand des BMBF. Karlsruhe (2004)
- LearnTec - Internationale Messe und Kongress für Bildungstechnologie. Messe-Präsentation MONIST auf dem BMBF-Messestand. Karlsruhe (2004)
- EuroCogSci - Europäischer Kongress zur Kognitionswissenschaft. Poster- und Technikpräsentation zu MONIST. Osnabrück (2003)
- World Education Market - Internationale Messe und Konferenz für Bildungstechnologie. Teilnahme, Vortrag und Messe-Demo zu RUBIN. Vancouver, Canada (2001)