

Das Essener-Lern-Modell (ELM): Ein Vorgehensmodell zur Entwicklung computerunterstützter Lernumgebungen

Dissertation zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften (Dr. rer. pol.)

durch den Fachbereich Wirtschaftswissenschaften der
Universität Essen

Vorgelegt von
Dipl.-Wirt. Inform. Jan Martin Pawlowski
aus Essen

Gutachter: 1. Prof. Dr. Heimo H. Adelsberger
2. Prof. Dr. Stefan Eicker

eingereicht: 29.11.2001
Disputation: 14.12.2001

Zusammenfassung

Das Essener-Lern-Modell (ELM) ist ein generisches Vorgehensmodell, das den Entwicklungsprozess von computerunterstützten Lernumgebungen unterstützt. Während Modelle der Softwareentwicklung, didaktische Modelle oder Domänen-spezifische Modelle nur Teilbereiche des Entwicklungsprozesses umfassen, unterstützt das Essener-Lern-Modell den Entwicklungsprozess von der Curriculumentwicklung bis zu Entwicklung von Lerneinheiten. Hinzu kommen Querschnittsfunktionen wie Projektmanagement, Prozessintegration und Qualitätssicherung. Das Essener-Lern-Modell integriert die Spezifikation und Verwendung von Lerntechnologiestandards wie LOM und SCORM. Zusätzlich wird eine Spezifikation zur Beschreibung des Anwendungskontextes und von didaktischen Modellen entwickelt, die in Verbindung mit bestehenden Standards ein Rahmenkonzept für die Interoperabilität von Lernmanagementsystemen und Lernumgebungen liefert. Die ELM-Applikation ist ein Werkzeug, das zur Unterstützung von Entwicklungsteams implementiert wurde und die Konzepte des Essener-Lern-Modells umsetzt. Aus den Konzepten und Spezifikationen werden von der ELM-Applikation computerunterstützte Lernumgebungen in XML generiert, die exemplarisch dargestellt werden.

Abstract

The Essen Learning Model is a generic development model, supporting developers, educators, and users on different levels of educational activities. Existing software engineering models, pedagogical models, and domain-specific models solely support parts of the development process. The Essen Learning Model provides concepts for project management, quality assurance, process integration, curriculum development, and the development of learning sequences and learning units. Emerging learning technology standards such as SCORM and LOM are integrated in the development process, minimizing the authors' specification effort. Additionally, a detailed specification for context and didactical models was developed. Combining these standards with existing learning technology standards leads to an interoperable framework for learning management systems and learning environments. Furthermore, the ELM application is a tool to support development processes. Using this tool, XML-based learning environments and specifications of learning technology standards are generated. Finally, learning environments, designed with ELM, are presented.

Schlagwörter:

Computerunterstütztes Lernen, Vorgehensmodell, Prozessmodellierung, Lernumgebungen, Standardisierung, Lerntechnologien, Didaktische Methoden, E-Learning

Keywords:

Computer Assisted Learning, Development Model, Process Model, Learning Environment, Standardization, Learning Technologies, Didactical Methods, E-Learning

Inhaltsverzeichnis

| | |
|---|-------------|
| ZUSAMMENFASSUNG..... | I |
| INHALTSVERZEICHNIS | II |
| ABBILDUNGSVERZEICHNIS..... | VI |
| TABELLENVERZEICHNIS | IX |
| ABKÜRZUNGSVERZEICHNIS..... | XI |
| VORWORT..... | XVII |
| 1 EINLEITUNG | 1 |
| 1.1 ZIELSETZUNG DER ARBEIT | 2 |
| 1.2 AUFBAU DER ARBEIT | 3 |
| 2 COMPUTERUNTERSTÜTZTE LERNUMGEBUNGEN..... | 5 |
| 2.1 BEGRIFFSABGRENZUNG | 5 |
| 2.2 KLASSIFIKATION VON LERNTECHNOLOGIEN | 7 |
| 2.2.1 <i>Funktionale Klassifikation</i> | <i>7</i> |
| 2.2.2 <i>Methodische Klassifikation.....</i> | <i>9</i> |
| 2.2.2.1 Hilfesysteme | 9 |
| 2.2.2.2 Lernergesteuerte Systeme | 10 |
| 2.2.2.3 Trainingssysteme | 11 |
| 2.2.2.4 Tutorielle Systeme | 12 |
| 2.2.2.5 Problemlösungssysteme..... | 13 |
| 2.2.2.6 Simulations- und Spielsysteme..... | 13 |
| 2.2.2.7 Kollaborative Lernumgebungen | 14 |
| 2.2.2.8 Kognitive Werkzeuge | 14 |
| 2.2.2.9 Kritische Würdigung..... | 14 |
| 2.3 BESCHREIBUNGSMERKMALE VON LERNUMGEBUNGEN | 15 |
| 2.3.1 <i>Kontext</i> | <i>17</i> |
| 2.3.1.1 Ebenen des didaktischen Handelns..... | 17 |
| 2.3.1.2 Organisationale Voraussetzungen..... | 18 |
| 2.3.2 <i>Lernsituation</i> | <i>19</i> |
| 2.3.2.1 Zeitliche Einordnung | 19 |
| 2.3.2.2 Räumliche Einordnung | 19 |
| 2.3.3 <i>Lernziele und Lerninhalte</i> | <i>19</i> |
| 2.3.3.1 Lernziele | 20 |
| 2.3.3.2 Lerninhalte | 22 |
| 2.3.4 <i>Lerntheorien.....</i> | <i>24</i> |
| 2.3.4.1 Behaviorismus | 24 |
| 2.3.4.2 Kognitivismus..... | 26 |
| 2.3.4.3 Konstruktivismus | 27 |
| 2.3.4.4 Zusammenfassung und Folgerungen | 29 |
| 2.3.5 <i>Aktoren.....</i> | <i>31</i> |

| | | |
|----------|---|-----------|
| 2.3.5.1 | Rolle..... | 31 |
| 2.3.5.2 | Lernende | 32 |
| 2.3.5.3 | Lehrende | 34 |
| 2.3.6 | <i>Interaktion</i> | 34 |
| 2.3.7 | <i>Präsentation</i> | 37 |
| 2.3.8 | <i>Didaktische Methoden</i> | 38 |
| 2.3.8.1 | Fallmethode (Fallstudie)..... | 39 |
| 2.3.8.2 | Fernunterricht (Fernkurs, Fernstudium, Korrespondenz-Unterricht) | 40 |
| 2.3.8.3 | Frontalunterricht (Klassenunterricht, Vorlesung)..... | 40 |
| 2.3.8.4 | Individueller Lernplatz (Lernecke, Selbstlernplatz) | 41 |
| 2.3.8.5 | Individualisierter Programmierter Unterricht, Programmierete Unterweisung | 42 |
| 2.3.8.6 | Simulations- und Rollenspiele | 43 |
| 2.3.8.7 | Kritische Würdigung..... | 44 |
| 2.3.9 | <i>Evaluation</i> | 45 |
| 2.3.10 | <i>Zusammenfassung</i> | 47 |
| 2.4 | ARCHITEKTUR COMPUTERUNTERSTÜTZTER LERNUMGEBUNGEN | 47 |
| 2.4.1 | <i>Architekturen von Lernumgebungen</i> | 48 |
| 2.4.2 | <i>Learning Technology Systems Architecture (L TSA)</i> | 51 |
| 2.4.3 | <i>Folgerungen</i> | 54 |
| 3 | VORGEHENSMODELLE | 55 |
| 3.1 | BEGRIFFSABGRENZUNG | 55 |
| 3.2 | BESCHREIBUNG UND BEWERTUNG VON VORGEHENSMODELLEN | 59 |
| 3.2.1 | <i>Beschreibungsmerkmale</i> | 59 |
| 3.2.2 | <i>Beschreibungsmatrix für Vorgehensmodelle</i> | 61 |
| 3.2.3 | <i>Bewertungskriterien für Vorgehensmodelle</i> | 63 |
| 3.2.3.1 | Allgemeine Kriterien der Softwareentwicklung | 63 |
| 3.2.3.2 | Spezifische Kriterien zur Entwicklung von computerunterstützten Lernumgebungen | 64 |
| 3.2.4 | <i>Zusammenfassung der Bewertungskriterien</i> | 70 |
| 3.3 | VORGEHENSSTRATEGIEN UND ALLGEMEINE VORGEHENSMODELLE..... | 72 |
| 3.3.1 | <i>Lineare Vorgehensstrategie</i> | 72 |
| 3.3.2 | <i>Iterative Vorgehensstrategie</i> | 73 |
| 3.3.3 | <i>Prototyping</i> | 74 |
| 3.3.4 | <i>Nebenläufige Vorgehensstrategie</i> | 74 |
| 3.3.5 | <i>Evolutionäre Vorgehensstrategien</i> | 75 |
| 3.3.6 | <i>Spiralmodell</i> | 76 |
| 3.3.7 | <i>Objektorientiertes Modell</i> | 77 |
| 3.4 | VORGEHENSMODELLE ZUR ENTWICKLUNG COMPUTERUNTERSTÜTZTER LERNUMGEBUNGEN | 79 |
| 3.4.1 | <i>Instruktionsdesign</i> | 79 |
| 3.4.2 | <i>Technologieorientierte Vorgehensmodelle für Lernumgebungen</i> | 83 |
| 3.4.2.1 | CBT-Vorgehensmodell..... | 83 |
| 3.4.2.2 | Hypermedia-Vorgehensmodelle | 84 |
| 3.4.3 | <i>Hybride Ansätze für Lernumgebungen</i> | 87 |
| 3.4.3.1 | Hybride Vorgehensmodelle | 87 |
| 3.4.3.2 | 3-Space-Design-Strategy | 88 |

| | | |
|----------|--|------------|
| 3.5 | FOLGERUNGEN..... | 89 |
| 4 | STANDARDISIERUNG VON LERNTECHNOLOGIEN..... | 90 |
| 4.1 | ZIELSETZUNG VON STANDARDS..... | 90 |
| 4.2 | STANDARDISIERUNG VON LERNTECHNOLOGIEN..... | 91 |
| 4.2.1 | <i>Korrespondierende Standards</i> | 92 |
| 4.2.1.1 | Dublin Core Metadata-Initiative..... | 92 |
| 4.2.1.2 | E-Business Standards..... | 94 |
| 4.2.1.3 | Administrative Standards..... | 95 |
| 4.2.2 | <i>Spezifische Standards</i> | 96 |
| 4.3 | EXTENSIBLE MARKUP LANGUAGE (XML)..... | 99 |
| 4.3.1 | <i>XML-Konzept</i> | 99 |
| 4.3.2 | <i>Technologie</i> | 100 |
| 4.3.3 | <i>Konzepte</i> | 101 |
| 4.4 | LERNTECHNOLOGIESTANDARDS..... | 103 |
| 4.4.1 | <i>Kategorisierung und Spezifikationsformate</i> | 103 |
| 4.4.2 | <i>Generelle Standards</i> | 104 |
| 4.4.3 | <i>Metadaten</i> | 104 |
| 4.4.3.1 | Learning Object Metadata (LOM)..... | 105 |
| 4.4.4 | <i>Lernerorientierte Standards</i> | 107 |
| 4.4.4.1 | Public and Private Information (PAPI)..... | 107 |
| 4.4.4.2 | Learner Information Package (LIP)..... | 109 |
| 4.4.4.3 | Reusable Competencies Definitions..... | 109 |
| 4.4.4.4 | Universal Learning Format..... | 109 |
| 4.4.5 | <i>Inhaltsorientierte Standards</i> | 110 |
| 4.4.5.1 | Content Packaging..... | 110 |
| 4.4.5.2 | Learning Material Markup Language..... | 111 |
| 4.4.6 | <i>Managementorientierte Standards</i> | 112 |
| 4.4.6.1 | Computer Managed Instruction (CMI)..... | 112 |
| 4.4.6.2 | Question and Test Interoperability (QTI)..... | 113 |
| 4.4.6.3 | Enterprise..... | 114 |
| 4.4.7 | <i>Didaktische Standards</i> | 114 |
| 4.5 | SHARABLE CONTENT OBJECT REFERENCE MODEL..... | 115 |
| 4.5.1 | <i>Content Aggregation Model</i> | 116 |
| 4.5.2 | <i>Run-Time Environment</i> | 118 |
| 4.5.3 | <i>Kritische Würdigung</i> | 119 |
| 4.6 | EDUCATIONAL MODELLING LANGUAGE..... | 119 |
| 4.7 | FOLGERUNGEN..... | 121 |
| 5 | KONZEPTION DES ESSENER-LERN-MODELLS..... | 122 |
| 5.1 | KRITERIEN ZUR ENTWICKLUNG VON ELM..... | 122 |
| 5.2 | VORGEHENSSTRATEGIE DES ESSENER-LERN-MODELLS..... | 124 |
| 5.3 | ARCHITEKTUR DES ESSENER-LERN-MODELLS..... | 126 |
| 5.4 | VORGEHENSMODELL ZUR AUSBILDUNGSPLANUNG (ELM-C)..... | 129 |
| 5.4.1 | <i>Projektinitiierung</i> | 130 |
| 5.4.2 | <i>Kontextanalyse</i> | 133 |
| 5.4.3 | <i>Curriculumdesign/Evaluation</i> | 136 |
| 5.4.4 | <i>Zusammenfassung ELM-C</i> | 139 |
| 5.5 | VORGEHENSMODELL ZUR ENTWICKLUNG VON LERNSEQUENZEN (ELM-D)..... | 139 |

| | | |
|----------|--|------------|
| 5.5.1 | <i>Wissensakquisition</i> | 141 |
| 5.5.2 | <i>Benutzeranalyse</i> | 143 |
| 5.5.3 | <i>Methodenmodellierung</i> | 145 |
| 5.5.4 | <i>Lernumgebungsdesign</i> | 149 |
| 5.5.5 | <i>Zusammenfassung ELM-D</i> | 149 |
| 5.6 | VORGEHENSMODELL ZUR ENTWICKLUNG VON LERNEINHEITEN (ELM-E)..... | 150 |
| 5.6.1 | <i>ELM-E: Ablauffestlegung</i> | 151 |
| 5.6.2 | <i>ELM-E: Präsentationsdesign</i> | 152 |
| 5.6.3 | <i>ELM-E: Interaktionsdesign</i> | 154 |
| 5.6.4 | <i>ELM-E: Evaluationsdesign</i> | 154 |
| 5.6.5 | <i>ELM-Run-Time</i> | 156 |
| 6 | UMSETZUNG DES ESSENER-LERN-MODELLS | 158 |
| 6.1 | BESCHREIBUNG DER ELM-APPLIKATION..... | 158 |
| 6.1.1 | <i>Entwicklungsentscheidungen</i> | 158 |
| 6.1.1.1 | <i>Programmiersprache</i> | 158 |
| 6.1.1.2 | <i>Technische Architektur</i> | 159 |
| 6.1.2 | <i>Applikationen</i> | 160 |
| 6.1.2.1 | <i>Anpassung</i> | 163 |
| 6.1.2.2 | <i>ELM-C</i> | 167 |
| 6.1.2.3 | <i>ELM-D</i> | 173 |
| 6.1.2.4 | <i>ELM-E</i> | 178 |
| 6.1.3 | <i>Metadaten-Navigator</i> | 183 |
| 6.2 | ANWENDUNGSBEISPIELE..... | 184 |
| 6.2.1 | <i>Prototypische Entwicklung einer Lernumgebung XML</i> | 185 |
| 6.2.1.1 | <i>Anpassung</i> | 185 |
| 6.2.1.2 | <i>ELM-C</i> | 185 |
| 6.2.1.3 | <i>ELM-D</i> | 190 |
| 6.2.1.4 | <i>ELM-E</i> | 197 |
| 6.2.2 | <i>Lernumgebung Grundlagen der Programmierung</i> | 203 |
| 6.2.2.1 | <i>Anpassung</i> | 203 |
| 6.2.2.2 | <i>ELM-C</i> | 203 |
| 6.2.2.3 | <i>ELM-D</i> | 206 |
| 6.2.2.4 | <i>ELM-E</i> | 209 |
| 6.2.3 | <i>MobiLum: Prototyp einer mobilen Lernumgebung</i> | 211 |
| 6.2.3.1 | <i>ELM-C</i> | 212 |
| 6.2.3.2 | <i>ELM-D</i> | 213 |
| 6.2.3.3 | <i>ELM-E</i> | 216 |
| 7 | ZUSAMMENFASSUNG UND AUSBLICK | 220 |
| 7.1 | ZUSAMMENFASSUNG..... | 220 |
| 7.2 | AUSBLICK..... | 222 |
| | LITERATURVERZEICHNIS | 224 |
| | LEBENS LAUF | 246 |
| | EIDESSTATTLICHE ERKLÄRUNG | 247 |

Abbildungsverzeichnis

| | |
|---|-----|
| ABBILDUNG 1: AUFBAU DER ARBEIT | 4 |
| ABBILDUNG 2: EINORDNUNG DER LT-SYSTEMKLASSEN | 8 |
| ABBILDUNG 3: HILFESYSTEME [BODE1990] | 9 |
| ABBILDUNG 4: LERNERGESTEUERTE SYSTEME [BODE1990]..... | 11 |
| ABBILDUNG 5: TRAININGSSYSTEME [BODE1990] | 11 |
| ABBILDUNG 6: TUTORIELLE SYSTEME [BODE1990]..... | 12 |
| ABBILDUNG 7: PROBLEMLÖSUNGSSYSTEME [BODE1990]..... | 13 |
| ABBILDUNG 8: SIMULATIONS-/SPIELSYSTEME [BODE1990]..... | 13 |
| ABBILDUNG 9: BERLINER MODELL [HESO1965]..... | 15 |
| ABBILDUNG 10: EBENEN DES DIDAKTISCHEN HANDELNS | 17 |
| ABBILDUNG 11: LERNZIELKONTINUUM..... | 20 |
| ABBILDUNG 12: HEURISTISCHES LEHR- UND LERNMODELL NACH [BAPA1994]..... | 23 |
| ABBILDUNG 13: BEISPIELE FÜR KOMMUNIKATIONSPROZESSE | 35 |
| ABBILDUNG 14: PRÄSENTATIONSFORMEN NACH [BOSc1995] | 37 |
| ABBILDUNG 15: ARCHITEKTUR NACH [FICK1992] | 48 |
| ABBILDUNG 16: ARCHITEKTUR INTEGRIERTER INTERNET-LERNUMGEBUNGEN [FEsc2001]..... | 49 |
| ABBILDUNG 17: ARCHITEKTUR VON LERNUMGEBUNGEN NACH [PAQU2001] | 50 |
| ABBILDUNG 18: LTSA-EBENEN [LTSC2001A] | 51 |
| ABBILDUNG 19: LTSA-SYSTEMKOMPONENTEN [LTSC2001A] | 52 |
| ABBILDUNG 20: ARCHITEKTUR ERILE [LIND2001B] | 53 |
| ABBILDUNG 21: BEZIEHUNGEN DER ABSTRAKTIONSEBENEN | 57 |
| ABBILDUNG 22: ORDNUNGSSCHEMA VORGEHENSMODELLE..... | 58 |
| ABBILDUNG 23: KRITERIEN NACH [DEKL2001]..... | 67 |
| ABBILDUNG 24: WASSERFALLMODELL | 72 |
| ABBILDUNG 25: ITERATIVE VORGEHENSSTRATEGIE | 73 |
| ABBILDUNG 26: PROTOTYPING | 74 |
| ABBILDUNG 27: NEBENLÄUFIGE VORGEHENSSTRATEGIE | 75 |
| ABBILDUNG 28: EVOLUTIONÄRE VORGEHENSSTRATEGIE..... | 76 |
| ABBILDUNG 29: SPIRALMODELL | 77 |
| ABBILDUNG 30: OBJEKTORIENTIERTE VORGEHENSSTRATEGIE | 78 |
| ABBILDUNG 31: TEILSYSTEME DER OOA [BALZ1996] | 78 |
| ABBILDUNG 32: EINFLUSSFAKTOREN DES INSTRUKTIONSDESIGNS | 80 |
| ABBILDUNG 33: PHASEN DES ID NACH [ISSI1997] | 80 |
| ABBILDUNG 34: CBT-VORGEHENSMODELL [STEP1989]..... | 83 |
| ABBILDUNG 35: ENTWICKLUNGSMODELL FÜR HYPERMEDIALE LERNSYSTEME [BLUM1998] | 86 |
| ABBILDUNG 36: ZUSAMMENHANG DER STANDARDISIERUNGSAKTIVITÄTEN..... | 96 |
| ABBILDUNG 37: ÜBERSICHT DER TOOLS IM PROJEKT ARIADNE [ARIA2000B] | 98 |
| ABBILDUNG 38: KENNZEICHNUNG EINES XML-ELEMENTES | 101 |
| ABBILDUNG 39: PAPI LEARNER INFORMATION [IEEE2000c]..... | 108 |

| | |
|--|-----|
| ABBILDUNG 40: IMS CONTENT PACKAGING..... | 111 |
| ABBILDUNG 41: INHALTSAGGREGATION IN SCORM [DODD2001A] | 116 |
| ABBILDUNG 42: SCORM RUN-TIME ENVIRONMENT [DODD2001A] | 118 |
| ABBILDUNG 43: ELM-VORGEHENSMODELL | 124 |
| ABBILDUNG 44: ELM-ENTWICKLUNGSEBENEN | 126 |
| ABBILDUNG 45: ELM-ARCHITEKTUR | 127 |
| ABBILDUNG 46: DATENSHEMA DER TOP-LEVEL-ENTITIES | 128 |
| ABBILDUNG 47: WERTSCHÖPFUNGSKETTE ELM-C | 130 |
| ABBILDUNG 48: DATENSHEMA ORGANISATION/KONTEXT | 134 |
| ABBILDUNG 49: DATENSHEMA CURRICULUM..... | 136 |
| ABBILDUNG 50: DATENSHEMA LERNUMGEBUNG..... | 140 |
| ABBILDUNG 51: WERTSCHÖPFUNGSKETTE ELM-D | 140 |
| ABBILDUNG 52: DATENSHEMA LERNOBJEKT | 142 |
| ABBILDUNG 53: DATENSHEMA METHODE | 146 |
| ABBILDUNG 54: WERTSCHÖPFUNGSKETTE ELM-E..... | 150 |
| ABBILDUNG 55: DATENSHEMA LERNEINHEIT | 151 |
| ABBILDUNG 56: ELM-RUN-TIME | 157 |
| ABBILDUNG 57: TECHNISCHE ARCHITEKTUR DER ELM-APPLIKATION..... | 159 |
| ABBILDUNG 58: TEILAPPLIKATION ELM-LOM | 161 |
| ABBILDUNG 59: TEILAPPLIKATION ELM-USER | 162 |
| ABBILDUNG 60: ELM-EDITOR | 162 |
| ABBILDUNG 61: TEILAPPLIKATION ELM-METHOD..... | 163 |
| ABBILDUNG 62: PROJEKTMANAGEMENT PROZESSANFORDERUNGEN | 165 |
| ABBILDUNG 63: PROJEKTMANAGEMENT PROZESSGRUPPEN | 165 |
| ABBILDUNG 64: PROJEKTMANAGEMENT LISTEN..... | 166 |
| ABBILDUNG 65: ELM-C PROJEKTBESCHREIBUNG | 167 |
| ABBILDUNG 66: ELM-C ZEITMANAGEMENT | 168 |
| ABBILDUNG 67: ELM-C UNTERNEHMENSANALYSE | 169 |
| ABBILDUNG 68: ELM-C AUSBILDUNGSANALYSE..... | 170 |
| ABBILDUNG 69: ELM-C LERNZIELE..... | 171 |
| ABBILDUNG 70: ELM-C LERNOBJEKTE | 172 |
| ABBILDUNG 71: ELM-C EVALUATION | 173 |
| ABBILDUNG 72: ELM-D QUELLENBEARBEITUNG..... | 174 |
| ABBILDUNG 73: ELM-D BENUTZERANALYSE | 175 |
| ABBILDUNG 74: ELM-D METHODENMODELLIERUNG..... | 176 |
| ABBILDUNG 75: ELM-D PHASEN EINER METHODE | 176 |
| ABBILDUNG 76: ELM-D PHASENBLÖCKE EINER METHODE..... | 177 |
| ABBILDUNG 77: EDITOR ZUR BEARBEITUNG VON LERNEINHEITEN | 179 |
| ABBILDUNG 78: ELM-E LERNEINHEITEN BEARBEITEN, MEDIENOBJEKT | 180 |
| ABBILDUNG 79: ELM-E LERNEINHEITEN BEARBEITEN, MEDIENGRUPPE..... | 181 |
| ABBILDUNG 80: ELM-E KOMMUNIKATION | 182 |
| ABBILDUNG 81: ELM-BERICHT | 183 |

| | |
|--|-----|
| ABBILDUNG 82: OBERFLÄCHE XML-LU | 196 |
| ABBILDUNG 83: ELM-E: PRÄSENTIEREN VON XML | 197 |
| ABBILDUNG 84: LERNZIELE <i>GRUNDLAGEN DER PROGRAMMIERUNG</i> | 205 |
| ABBILDUNG 85: LERNOBJEKTE <i>GRUNDLAGEN DER PROGRAMMIERUNG</i> | 208 |
| ABBILDUNG 86: NAVIGATION UND LAYOUT <i>GRUNDLAGEN DER PROGRAMMIERUNG</i> | 209 |
| ABBILDUNG 87: BEISPIEL EINER PRÜFUNGSAUFGABE | 210 |
| ABBILDUNG 88: LERNZIELE <i>XELE</i> | 213 |
| ABBILDUNG 89: METHODENMODELL DES MOBILLEN PLANSPIELS | 215 |
| ABBILDUNG 90: LERNOBJEKTE <i>XELE</i> | 216 |
| ABBILDUNG 91: SCHEMATISCHE ÜBERSICHT DER BILDSCHIRMGESTALTUNG | 218 |
| ABBILDUNG 92: LAYOUT MOBILUM | 218 |
| ABBILDUNG 93: MOBILUM-ARCHITEKTUR | 219 |

Tabellenverzeichnis

| | |
|---|-----|
| TABELLE 1: ZUSAMMENFASSUNG DER LERNTHEORIEN [BLUM1998] | 29 |
| TABELLE 2: ROLLEN IN LERNPROZESSEN | 31 |
| TABELLE 3: LERNTYPEN NACH [KOLB1981] | 33 |
| TABELLE 4: ROLLEN DER LEHRENDEN | 34 |
| TABELLE 5: PHASEN DER FALLMETHODE | 39 |
| TABELLE 6: PHASEN DES FERNUNTERRICHTS | 40 |
| TABELLE 7: PHASEN DES FRONTALUNTERRICHTS | 41 |
| TABELLE 8: PHASEN DES INDIVIDUELLEN LERNPLATZES | 42 |
| TABELLE 9: PHASEN DER PROGRAMMIERTEN UNTERWEISUNG | 42 |
| TABELLE 10: PHASEN DES SIMULATIONSSPIELS | 43 |
| TABELLE 11: DETERMINANTEN DER ENTWICKLUNG VON LERNUMGEBUNGEN | 47 |
| TABELLE 12: BESCHREIBUNGSMATRIX FÜR VGM | 62 |
| TABELLE 13: QUALITÄTSKRITERIEN NACH [KUGE2001] | 69 |
| TABELLE 14: BESCHREIBUNGSMATRIX FÜR VGM ZUR ENTWICKLUNG VON LERNUMGEBUNGEN | 71 |
| TABELLE 15: PERFORMANCE-CONTENT MATRIX NACH [MERR1994] | 81 |
| TABELLE 16: DUBLIN CORE-METADATEN | 93 |
| TABELLE 17: GESCHÄFTSPROZESSE INNERHALB VON EBXML [EBXM2000] | 95 |
| TABELLE 18: ZUSAMMENFASSUNG KOOPERIERENDER STANDARDS UND APPLIKATIONEN . | 100 |
| TABELLE 19: LEARNING OBJECT METADATA | 106 |
| TABELLE 20: KATEGORIEN DER PUBLIC AND PRIVATE LEARNER INFORMATION | 108 |
| TABELLE 21: GEGENÜBERSTELLUNG PAPI - LIP | 109 |
| TABELLE 22: KATEGORIEN DER COMPUTER MANAGED INSTRUCTION | 113 |
| TABELLE 23: OBJEKTE DES QTI | 114 |
| TABELLE 24: CONTENT STRUCTURE FORMAT | 117 |
| TABELLE 25: INFORMATIONSMODELL EML | 120 |
| TABELLE 26: INFORMATIONSMODELL PROJEKTDATEN | 132 |
| TABELLE 27: INFORMATIONSMODELL KONTEXTANALYSE | 135 |
| TABELLE 28: INFORMATIONSMODELL CURRICULUM | 138 |
| TABELLE 29: INFORMATIONSMODELL LERNOBJEKT | 143 |
| TABELLE 30: INFORMATIONSMODELL AKTOREN | 144 |
| TABELLE 31: INFORMATIONSMODELL METHODE | 148 |
| TABELLE 32: LMML-TYPEN | 152 |
| TABELLE 33: INFORMATIONSMODELL LERNEINHEIT | 155 |
| TABELLE 34: IT-MINDESTANFORDERUNGEN FÜR AUTOREN UND ANWENDER | 187 |
| TABELLE 35: LERNZIELMATRIX XML | 188 |
| TABELLE 36: SPEZIFIKATION DER XML RECOMMENDATION | 191 |
| TABELLE 37: EXEMPLARISCHE ZUORDNUNG VON RESSOURCEN ZU LERNZIELEN | 192 |
| TABELLE 38: DETERMINIERUNG DER UNTERRICHTSMETHODEN | 194 |

| | |
|---|-----|
| TABELLE 39: ABLAUF DES XML-KURSES..... | 198 |
| TABELLE 40: ELM-E AM BEISPIEL FALLSTUDIE..... | 202 |
| TABELLE 41: INHALTE VON VAWI..... | 204 |

Abkürzungsverzeichnis

| | |
|---------|---|
| ADL | Advanced Distributed Learning |
| ADLNET | Advanced Distributed Learning Network |
| AICC | Aviation Industry CBT Committee |
| API | Application Programming Interface |
| ARIADNE | Alliance of Remote Instructional Authoring Distribution Networks for Europe |
| ARIS | Architektur integrierter Informationssysteme |
| ASP | Active Server Pages |
| bez. | bezüglich |
| BMEcat | Bundesverband für Materialwirtschaft, Einkauf und Logistik-Katalog |
| bzw. | beziehungsweise |
| CASE | Computer Aided Software Engineering |
| CDT | Component Display Theory |
| CEN | European Committee for Standardization |
| CGI | Common Gateway Interface |
| CMI | Computer Managed Instruction |
| CML | Chemical Markup Language |
| CORBA | Common Object Request Broker Architecture |
| CSCW | Computer Supported Cooperative Work |
| CSF | Content Structure Format |
| CSS | Cascading Style Sheets |
| CXML | Commerce eXtensible Markup Language |
| DC | Dublin Core |
| d. h. | das heißt |
| DIN | Deutsches Institut für Normung |
| DoD | Department of Defense |
| DOM | Document Object Model |

| | |
|------------|---|
| DSSSL | Document Style Semantics and Specific Language |
| DTD | Dokumenttyp-Definition |
| EBXML | Electronic Business Markup Language |
| ECML | Electronic Commerce Markup Language |
| EDI | Electronic Data Interchange |
| EEM | Electronic Education Market |
| E-Learning | Electronic Learning |
| ELM | Essener-Lern-Modell |
| ELM-C | C-Ebene des Essener-Lern-Modells |
| ELM-D | D-Ebene des Essener-Lern-Modells |
| ELM-E | E-Ebene des Essener-Lern-Modells |
| ELM-RT | ELM-Run-Time |
| E-Mail | Electronic Mail |
| EML | Educational Modelling Language |
| EPK | Ereignisgesteuerte Prozesskette |
| EPSS | Electronic Performance Support System |
| ERILE | Expertise and Role Identification for Learning Environments |
| etc. | et cetera |
| e.V. | eingetragener Verein |
| FAQ | Frequently Asked Questions |
| FTP | File Transfer Protocol |
| GB | Gigabyte |
| GSM | Globales System für Mobilkommunikation |
| GUI | Graphical User Interface |
| HDM | Hypertext Design Model |
| HR | Human Resources |
| HTML | Hypertext Markup Language |
| HTTP | Hypertext Transfer Protocol |
| ID | Instruktionsdesign |

| | |
|--------------------|---|
| ID ₂ | Instructional Design der zweiten Generation |
| IEC | International Electrotechnical Commission |
| IEEE | Institute of Electrical and Electronics Engineers |
| IMDL | Instructional Material Design Language |
| IMS | Instructional Management Systems |
| ISDN | Integrated Services Digital Network |
| ISO | International Organization for Standardization |
| ISSS | Information Society Standardization System |
| IT | Informationstechnologie |
| JSML | Java Speech Markup Language |
| JTC | Joint Technical Committee |
| KBit | Kilobit |
| LDAP | Lightweight Directory Access Protocol |
| LIP | Learner Information Package |
| LMML | Learning Material Markup Language |
| LOM | Learning Object Metadata |
| LT | Lerntechnologien, Learning Technologies |
| LTSA | Learning Technology Systems Architecture |
| LTSC | Learning Technology Standards Committee |
| MathML | Mathematical Markup Language |
| MBit | Megabit |
| MB | Megabyte |
| MHz | Megahertz |
| M-Learning | Mobile Learning |
| MLU | Mobile Lernumgebung |
| OOA | Objektorientierte Analyse |
| OOD | Objektorientiertes Design |
| OSTP | White House Office of Science and Technology Policy |
| PaKM _{as} | Passauer Knowledge Management System |

| | |
|----------|---|
| PAPI | Public And Private Information |
| PC | Personal Computer |
| PDF | Portable Document Format |
| QK | Qualitätskriterium |
| QM | Qualitätsmanagement |
| QS | Qualitätssicherung |
| QTI | Question and Test Interoperability |
| RDF | Resource Description Framework |
| RMM | Relationship Management Methodology |
| s | Sekunde |
| SCO | Sharable Content Object |
| SCORM | Sharable Content Object Reference Model |
| SEE | Simulation Game on Educational Experiences |
| SIF | Schools Interoperability Framework Implementation Specification |
| SGML | Standard Generalized Markup Language |
| SMIL | Synchronized Multimedia Markup Language |
| SMS | Short Message Service |
| SMTP | Simple Mail Transfer Protocol |
| SpeechML | Speech Markup Language |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TML | Tutorial Modelling Language |
| TUM | Technische Universität München |
| u. a. | unter anderem |
| ULF | Unified Learning Format |
| UMTS | Universal Mobile Telecommunication System |
| URI | Unified Resource Identifier |
| URL | Uniform Resource Locator |
| VAWi | Virtuelle Aus- und Weiterbildung Wirtschaftsinformatik |
| vgl. | vergleiche |

| | |
|--------|--|
| VGM | Vorgehensmodell |
| VGS | Vorgehensstrategie |
| VXML | Voice eXtensible Markup Language |
| W3C | World Wide Web Consortium |
| WAP | Wireless Application Protocol |
| WBT | Web-based Training |
| WiP | Wirtschaftsinformatik der Produktionsunternehmen |
| WWW | World Wide Web |
| XELE | XML-based Explorative Learning Environment |
| XHTML | Extensible Hypertext Markup Language |
| XLL | Extensible Linking Language |
| XLink | Extensible Linking Language |
| XML-LU | XML-basierte Lernumgebung |
| XML | Extensible Markup Language |
| XQL | XML Query Language |
| XSL | Extensible Stylesheet Language |
| z. B. | zum Beispiel |

F ü r A lice

Vorwort

Die vorliegende Arbeit entstand im Fachgebiet Wirtschaftsinformatik der Produktionsunternehmen der Universität Essen im Zeitraum von 1997 bis 2001. Der Beginn dieser Arbeit war der Wunsch, einen Forschungsbereich zu erschließen, der auf die Kompetenzen des Studiums und des Fachgebietes aufsetzt, den persönlichen Interessen entspricht und einen positiven Beitrag zur gesellschaftlichen Entwicklung leistet.

Dabei zeigte sich sehr schnell, dass im Bereich des E-Learning unerschöpfliche Möglichkeiten der Forschung und Entwicklung bestehen. Die Forschungsarbeiten identifizierten jedoch schon in der ersten Analysephase Schwächen und Unzulänglichkeiten bestehender Modelle und Systeme. Daraus resultierte die Notwendigkeit und Aufgabe dieser Arbeit, ein umfassendes Vorgehensmodell zu entwickeln, das technologische, didaktische, wirtschaftliche Aspekte abdeckt. Die vorliegende Arbeit soll als Leitfaden zur Entwicklung computerunterstützter Lernumgebungen dienen. So umfasst das Essener-Lern-Modell ein breites Anwendungsfeld, beginnend mit der Curriculumentwicklung bis hin zur Gestaltung einzelner Lerneinheiten. Hinzu kommen Unterstützungsfunktionen für das Projektmanagement und Qualitätssicherung. Ein weiterer Schwerpunkt liegt auf der Integration von Standards im Bereich der Lerntechnologien.

Die Komplexität aber auch Faszination dieses Forschungsbereiches macht die Interdisziplinarität aus: die Verbindung pädagogischer, psychologischer und informationstechnologischer Perspektiven, die Integration komplexer Abläufe und nicht zuletzt die Teamarbeit mit Beteiligten unterschiedlichster Kontexte. Im Vordergrund steht für mich persönlich die Möglichkeit, Aus- und Weiterbildung zu gestalten und dadurch die berufliche und persönliche Entwicklung der Lernenden zu verbessern. So kann die Disziplin der Wirtschaftsinformatik positiv zur Entwicklung der Gesellschaft beitragen.

Die erste Danksagung gebührt meiner Frau Alice Pawlowski, die mir immer wieder Rückhalt und Bestätigung gegeben hat, auf dem richtigen Weg zu sein und nicht zuletzt immer wieder für die Ausgewogenheit von Wissenschafts- und Privatleben gesorgt hat. Mein Dank gilt meinen Eltern, Elke und Antonius Pawlowski, die meine akademische Laufbahn erst ermöglicht und schon früh die notwendige Neugier und Kreativität gefördert haben, ohne die eine wissenschaftliche Laufbahn nicht möglich wäre.

Diese Arbeit wäre nicht ohne Herrn Prof. Dr. Heimo H. Adelsberger entstanden, der meine Arbeit über den gesamten Zeitraum mit Begeisterung und der notwendigen Skepsis unterstützt und gefördert und mir den nötigen Freiraum zur Entwicklung der Dissertation gewährt hat. Ebenso bedanke ich mich bei Prof. Dr. Stefan Eicker für die Übernahme des Zweitgutachtens.

Besonders bedanken möchte ich mich bei meinen Kollegen, Frank Körner und Markus Bick, die mir nicht nur fachlich und als Freunde zur Seite standen, sondern eine Atmosphäre geschaffen haben, in der man einfach gerne arbeitet. Herrn Patrick Veith danke ich für seine begeisterte Mitarbeit und seine technische Kompetenz. Herrn Thomas Hanke danke

ich für die kommunikationswissenschaftliche Beratung, allen meinen weiteren Kollegen und studentischen Mitarbeitern für die allgegenwärtige Hilfsbereitschaft und Kooperation.

Für seine unermüdliche Programmiertätigkeit und seine ansteckende Gelassenheit möchte ich Herrn Dogan Azkan herzlich danken.

Die Ergebnisse der Arbeit ermutigen zu weiteren Forschungsaktivitäten in verschiedensten Teilbereichen des computerunterstützten Lernens. Ich hoffe, mit dieser Arbeit Potenziale aufzeigen und das Lernen der Zukunft mitgestalten zu können.

Essen, im November 2001

Jan M. Pawlowski

1 Einleitung

Die wirtschaftliche und informationstechnologische Entwicklung hat in den letzten Jahren eine dramatische Veränderung erfahren. Trends wie die Entwicklung zur Informations- und Kommunikationsgesellschaft, die Globalisierung, die Konzentration auf Kernkompetenzen und die damit verbundenen organisatorischen Veränderungen und nicht zuletzt die Arbeitsmarktsituation haben die Anforderungen an Mitarbeiter in Unternehmen drastisch verändert: Aufgrund der Entwicklungen in den Bereichen Organisation und Informationstechnologie werden sich Berufstätige mehr denn je kontinuierlich weiterbilden müssen, um ihre Beschäftigungsfähigkeit zu erhalten und auszubauen. Das traditionelle Konzept der einmaligen Berufs- bzw. Hochschulausbildung („Lernen auf Vorrat“) wird in Zukunft von flexibleren Bildungsformen abgelöst, oft zusammengefasst als *Lifelong Learning*.

Aus der Berücksichtigung von zeitlichen Restriktionen, denen Arbeitnehmer unterliegen, sowie wirtschaftlicher Interessen der Arbeitgeber entwickeln sich ein steigender Bedarf und zunehmende Akzeptanz computerunterstützten Lernens. In diesem Zusammenhang wird der Einsatz neuer Medien oft als Lösung aller Herausforderungen im Ausbildungssektor gesehen. Die Virtuelle Universität, Multimedia oder Edutainment sind nur einige Schlagworte, die sich in den letzten Jahren in Forschung und Praxis durchgesetzt haben. Eine aktuelle Studie der Berlecon Research erwartet ein durchschnittliches Wachstum der E-Learning-Branche um 50 Prozent und in Deutschland bis zum Jahr 2005 ein Marktvolumen von 1,5 bis 2 Milliarden Euro [Berl2001]. Die Entwicklungen im Bereich des computerunterstützten Lernens liefern allerdings nicht nur positive Lösungsansätze: Neben Innovationen und positiven Entwicklungen sind auch wenig ausgereifte oder ungeeignete Produkte und Konzepte zu finden. Gerade im Rahmen der Ausbildung ist die Qualitätssicherung noch nicht zum Standard geworden.

Die Verwendung von Vorgehensmodellen ist ein erster Schritt zu einer systematischen Qualitätssicherung. Es gibt jedoch einige Einschränkungen bezüglich der erfolgreichen Anwendung solcher Modelle: Vorgehensmodelle zur Softwareentwicklung sind aufgrund des fehlenden Bezuges zu didaktischen Konzepten nur bedingt geeignet, den Entwicklungsprozess angemessen zu unterstützen. Dagegen setzen spezifische Vorgehensmodelle, die zur Entwicklung von CBT-Systemen oder Hypermedia-Anwendungen genutzt werden, eine Auswahl einer bestimmten Systemklasse (z. B. Hypermedia, CBT, WBT) und somit der didaktischen Methoden bereits voraus, ohne die Anwendbarkeit im betrachteten Kontext zu evaluieren. Es wird deutlich, dass existierende Modelle nur Teilbereiche der Entwicklung von Lernumgebungen umfassen oder nur bestimmte Perspektiven (Technologiezentrierung, Didaktikzentrierung) auf den Entwicklungsprozess berücksichtigen. Nur wenige Modelle beschreiben einen Entwicklungsprozess, der technologische wie didaktische Aspekte adäquat einbezieht.

Ziel dieser Arbeit ist die Entwicklung des *Essener-Lern-Modells (ELM)* – eines *generischen Vorgehensmodells*, das die Qualität des Entwicklungsprozesses von Lernumgebungen auf verschiedenen Ebenen in didaktischer, wirtschaftlicher und fachlicher Hinsicht sicherstellt. Es unterstützt Design- und Entwicklungsprozesse durchgängig von der Curriculumentwicklung bis zur Umsetzung einzelner Lerneinheiten.

Im Folgenden werden zunächst die Zielsetzung und die Randbedingungen der Arbeit verdeutlicht, um die Aufgabenstellung eindeutig abzugrenzen. Des Weiteren wird die Methodik der Arbeit und der daraus resultierende Aufbau erläutert.

1.1 Zielsetzung der Arbeit

Ziel der Arbeit ist die Erstellung eines umfassenden Vorgehensmodells für Planung, Design, Entwicklung, Umsetzung und Wartung computerunterstützter Lernumgebungen. Die Teilbereiche der Aufgabenstellung werden zum besseren Verständnis genauer erläutert.

Das Vorgehensmodell soll eine *umfassende Planung und Konzeption* der für Entwicklungsprojekte in Unternehmen wie Institutionen relevanten didaktischen Ebenen ermöglichen. Bestehende Vorgehensmodelle betrachten nur Teilbereiche des Entwicklungsprozesses computerunterstützter Lernumgebungen, wie z. B. die Entwicklung hypermedialer Lerneinheiten. Daher muss zunächst der Umfang des zu entwickelnden Modells bestimmt werden.

Es soll ein *generisches Vorgehensmodell* entwickelt werden, das bestehende Modelle integriert und unabhängig vom Anwendungskontext einsetzbar ist sowie durch neue Modelle oder Richtlinien erweitert werden kann. Die Vielzahl möglicher Ausprägungen computerunterstützter Lernumgebungen (siehe Kapitel 2) hat eine mindestens ebenso große Anzahl an Vorgehensmodellen und Richtlinien hervorgebracht, die jedoch häufig nur Teilaspekte einbeziehen oder eine eingeschränkte Sichtweise auf das Anwendungsfeld haben.

Das Vorgehensmodell soll in *bestehende Geschäftsprozessmodelle* integrierbar sein. In der Praxis sind Geschäftsprozesse und Lernprozesse weitgehend voneinander getrennt [ReRa2001]. Diese Trennung führt zu ineffizienten Prozessen, da keine genaue Vergleichsmöglichkeit zwischen Bedarf und Angebot spezifischer Kompetenzen gegeben ist. Daher wird eine Integrationsmöglichkeit von Geschäfts- und Ausbildungsprozessen entwickelt.

Daneben soll das Vorgehensmodell die Spezifikation von *Lerntechnologiestandards* einbeziehen. In den letzten Jahren wurde eine Vielzahl von Standards im Bereich von Lerntechnologien spezifiziert. In dieser Arbeit werden relevante Standards identifiziert und im Vorgehensmodell berücksichtigt. Als Ergebnis der Planungsphase stehen Spezifikationen zur Verfügung, die diese Standards einbeziehen. Hinzu kommt die Anforderung, dass standardkonforme Beschreibungen (wie zum Beispiel die Beschreibung von Lernobjekten durch Metadaten) zu einem integralen Bestandteil des Entwicklungsprozesses werden.

Die *Ergebnisse* (z. B. Spezifikationen, Prototypen, Lernumgebung) der einzelnen Aktivitäten des Modells sollen in *interoperabler, adaptierbarer Form* zur Verbesserung der Wiederverwendbarkeit vorliegen. In den einzelnen Aktivitäten des Vorgehensmodells werden verschiedene Spezifikationen für Projektbeteiligte (z. B. Entwickler, Projektverantwortliche, Lernende) generiert. Diese sollen für die Zielgruppe individuell anpassbar und system- und anwendungsunabhängig nutzbar sein.

Es soll ein *Werkzeug* entwickelt werden, das Entwicklungsprozesse des Essener-Lern-Modells unterstützt. Unter Verwendung dieses Werkzeuges sollen computerunterstützte und insbesondere internetbasierte Lernumgebungen implementiert und evaluiert werden.

1.2 Aufbau der Arbeit

In diesem Abschnitt werden der Aufbau und das methodische Vorgehen der Arbeit erläutert. Die Vielschichtigkeit der Aufgabenstellung erfordert eine umfangreiche Analyse, Bewertung und Synthese bestehender Ansätze aus verschiedenen Disziplinen.

Im zweiten Kapitel werden die Grundlagen *computerunterstützter Lernumgebungen* dargestellt. Aufgrund der uneinheitlichen Terminologie in diesem Forschungsbereich werden zunächst die verwendeten Begriffe eindeutig definiert. Danach werden bestehende Klassifikationsansätze von Lernumgebungen vorgestellt und darauf aufbauend ein eigenes Klassifikationsschema entwickelt. Dies dient zur Beschreibung des Kontextes, zur Verdeutlichung der Einflussfaktoren auf Design- und Entwicklungsprozesse und zum Vergleich computerunterstützter Lernumgebungen.

Das dritte Kapitel der Arbeit befasst sich mit *Vorgehensmodellen*. Durch die Interdisziplinarität der Aufgabenstellung ist es offensichtlich, dass Vorgehensmodelle aus verschiedenen Domänen Anwendung finden können. Generische Vorgehensstrategien, Softwareentwicklungsmodelle, didaktische Modelle und hybride Ansätze werden betrachtet und auf ihre Eignung bezüglich der Entwicklung von Lernumgebungen untersucht. Die Analyse dieser Modelle identifiziert Stärken und Schwächen der Modelle und führt zu einer Bewertung. Ergebnis der Analyse ist die Formulierung von Anforderungen an ein Vorgehensmodell zur Entwicklung computerunterstützter Lernumgebungen.

Das vierte Kapitel erörtert die Anwendung von *Standards*: In den letzten Jahren wurden in verschiedenen Initiativen Standards für Lerntechnologien entwickelt, um z. B. den Inhalt von Lernressourcen oder Benutzercharakteristika zu beschreiben und interoperabel verfügbar zu machen. In diesem Kapitel wird diskutiert, welche Standards derzeit sinnvoll angewendet werden können und in welcher Form Standards innerhalb des Vorgehensmodells verwendet werden können. Es zeigt sich, dass gerade im Bereich der Beschreibung didaktischer Methoden noch keine adäquaten Ansätze vorhanden sind.

Im fünften Kapitel wird das Konzept des *Essener-Lern-Modells* im Detail erläutert. Basierend auf den Erkenntnissen und den Folgerungen der vorhergehenden Kapitel werden zunächst die Anforderungen an das Vorgehensmodell beschrieben und eine Architektur entwickelt. Es wird gezeigt, wie Design- und Entwicklungsprozesse unterstützt werden. Die Ebenen des Essener-Lern-Modells wurden nach den Ebenen didaktischen Handelns [Flehsig1996] benannt. Das Vorgehensmodell wird für die C-Ebene (Projektmanagement, Curriculumentwicklung), D-Ebene (Entwicklung von Lernsequenzen) und E-Ebene (Entwicklung von Lerneinheiten) entwickelt. Insbesondere wird gezeigt, wie die Spezifikation von Lerntechnologiestandards in den Entwicklungsprozess integriert werden kann.

Im sechsten Kapitel wird die *ELM-Applikation* vorgestellt. Diese Applikation ist ein java-basiertes Werkzeug, das Entwickler, Projektmanager, Lehrende und Lernende bei der Konzeption, Entwicklung und Implementierung von Lernumgebungen unterstützt. Mithilfe dieses Tools können umfangreiche Planungen vorgenommen und Lernumgebungen in der Extensible Markup Language (XML) generiert werden. Anhand verschiedener Kurse des Fachgebiets *Wirtschaftsinformatik der Produktionsunternehmen* und des Projekts *Virtuelle Aus- und Weiterbildung Wirtschaftsinformatik* wird die Anwendung dieses Tools exemplarisch gezeigt.

Die Arbeit schließt mit einer *Zusammenfassung* und einem *Ausblick*. Die Ergebnisse werfen neue Forschungspotentiale auf, die in Folgeprojekten bearbeitet werden können.

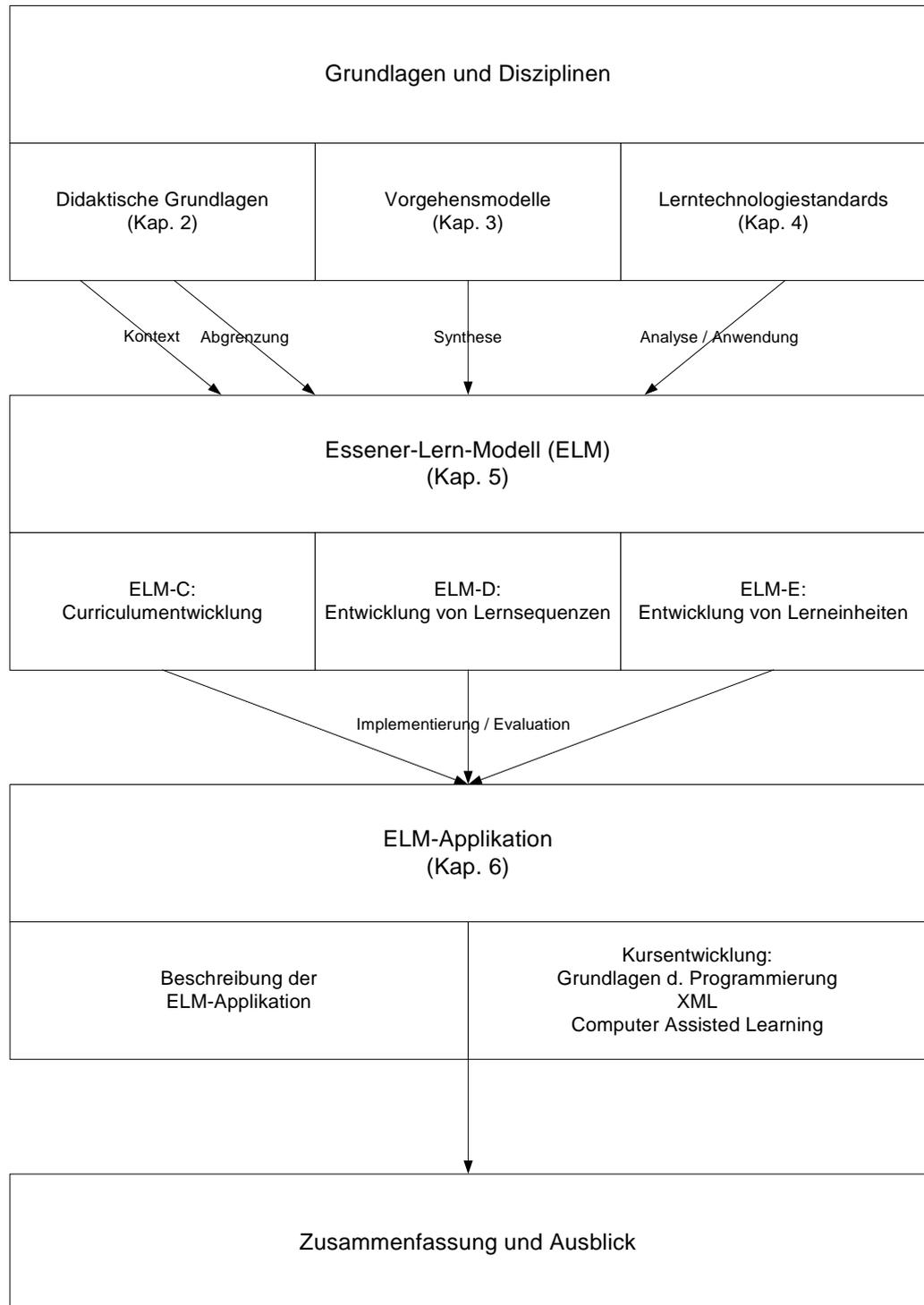


Abbildung 1: Aufbau der Arbeit

2 Computerunterstützte Lernumgebungen

Im folgenden Kapitel werden die für die Entwicklung des Vorgehensmodells benötigten didaktischen Grundlagen erläutert. Dabei werden zwei der zuvor genannten Zielsetzungen erfüllt: Einerseits wird der Anwendungskontext des Essener-Lern-Modells abgegrenzt. Andererseits werden die didaktischen Entscheidungsfelder erläutert, die im Rahmen des Entwicklungsprozesses computerunterstützter Lernumgebungen von Bedeutung sind. Durch die Erläuterung verschiedener Klassen von Lernumgebungen werden die Chancen und Möglichkeiten des computerunterstützten Lernens hervorgehoben; weiterhin werden dadurch Design- und Entwicklungsprozesse charakterisiert.

2.1 Begriffsabgrenzung

Dieser Abschnitt dient der Erläuterung der verwendeten Terminologie. Der Begriff des Lernens, insbesondere unter Verwendung von Informations- und Kommunikationssystemen, ist nicht eindeutig definiert. Verschiedene Disziplinen (z. B. Pädagogik, Psychologie, Informatik) und demzufolge differierende Schwerpunkte und Zielsetzungen haben zu einer heterogenen Begriffswelt geführt, die keine eindeutigen Abgrenzungen zulässt. Ziel dieses Kapitels ist eine Eingrenzung der Begriffe des computerunterstützten Lernens, ohne dabei eine Lerntheorie, eine Disziplin oder einen Anwendungsschwerpunkt in den Vordergrund zu stellen. Die Abgrenzungen werden kritisch diskutiert; daraus folgend wird für den weiteren Verlauf der Arbeit eine einheitliche Terminologie abgeleitet.

Nach [Gagn1980] ist *Lernen* „[...] eine Veränderung der menschlichen Disposition oder Fähigkeit, die über einen bestimmten Zeitraum erhalten bleibt und nicht durch Wachstumsprozesse zu erklären ist.“ [GaBe1996] definieren *Lernen* als „[...] Prozeß, durch den ein Organismus sein Verhalten als Resultat von Erfahrung ändert.“ Diese Definitionen mit pädagogischem Fokus zeigen, dass Lernen nicht als kurzfristige Aktivität gesehen werden kann, sondern ein komplexer Prozess ist, der zu Veränderungen des Verhaltens führt. Alle Aktivitäten, die zu dieser Verhaltensänderung beitragen, werden als *Lernprozesse* bezeichnet. Dabei umfassen Lernprozesse im umfassenden Sinne alle beteiligten Akteure, also neben den Lernenden auch Lehrende oder Tutoren.

Das Konzept des Lernens umfasst die Konzepte des Wissenserwerbs und der Wissensvermittlung, die sich auf kognitive Konzepte beziehen. Nach [ReMa1994] umfasst *Wissenserwerb* „[...] sowohl den Aufbau neuer Wissensstrukturen als auch die Anreicherung und Verfeinerung sowie Umstrukturierung bestehender Wissensstrukturen. *Wissensvermittlung* bezieht sich auf Ansätze und Methoden zur Anregung, Förderung und Unterstützung der verschiedenen Prozesse im Rahmen des Wissenserwerbs.“ [ReMa1994] betonen weiterhin, dass Lernen und Wissenserwerb nicht als einfacher Transport aus dem Kopf eines Lehrenden in den Kopf eines Lernenden gesehen werden kann, sondern dass es das Ergebnis komplexer Konstruktionsprozesse einzelner Individuen ist.

Der Einsatz von Computern zur Unterstützung von Lernprozessen muss ebenfalls differenziert betrachtet werden. Dies drückt sich ebenfalls in unterschiedlichen Begriffen aus. So wurden im englischsprachigen Raum die Begriffe *Computer Assisted Learning*, *Com-*

puter Based Learning, Computer Supported Learning, Computer Based Training, Computer Assisted Teaching, Computer Managed Instruction, Computer Assisted Instruction oder *Computer Based Education* geprägt, die den Einsatz computerunterstützter Unterrichtsmethoden in Lernprozessen beschreiben, aber bereits Annahmen über die Ausgestaltung der Lernprozesse implizieren. Im Deutschen drücken ebenfalls begriffliche Unterschiede bereits Annahmen über den Lernprozess aus. So deutet *computergestützt* einen wesentlich stärkeren Fokus bezüglich des Rechnereinsatzes als *computerunterstützt* an. *Computerunterstützte Lehre* weist weiterhin eine wesentlich stärkere Lehrerzentrierung als *computerunterstütztes Lernen* auf [vgl. Fick1992]. Da durch die verwendeten Begriffe keine Annahmen über den Lernprozess getroffen werden sollen, werden im Folgenden die wichtigsten in der Arbeit verwendeten Begriffe unabhängig von ihrer historischen Entstehung definiert.

Computerunterstütztes Lernen (CUL, Computer Supported Learning) bezeichnet die konzeptionellen Grundlagen und Methoden als Basis des Computereinsatzes in Lernprozessen.

Diese Definition fasst als Oberbegriff die Methoden und Konzepte zusammen, ohne Annahmen über die Ausgestaltung von Lernprozessen zu treffen. Lernen wird dabei als interaktiver Prozess der Beteiligten gesehen (Lernende, Lehrende, Computer). Der Begriff *E-Learning* wird in aktueller Literatur häufig als Schlagwort verwendet, kann jedoch synonym zu CUL angesehen werden.

Eine *computerunterstützte Lernumgebung (CULU, Computer Supported Learning Environment)* ist die Realisierung und Anwendung des computerunterstützten Lernens durch rechnergestützte Systeme, wobei die konzeptionellen und methodologischen Grundlagen des CUL umgesetzt werden.

Der Begriff der computerunterstützten Lernumgebung ist somit die Generalisierung einer Vielzahl von Instanzen von Informationssystemen. Da in dieser Arbeit ausschließlich computerunterstützte Lernformen betrachtet werden, wird der Begriff *Lernumgebung (LU)* als Verkürzung des oben genannten Begriffes synonym verwendet. Zusätzlich sei der Begriff der internetunterstützten Lernumgebung erwähnt:

Eine *internetunterstützte Lernumgebung (IULU)* ist eine Lernumgebung, die den Lernprozess durch unterschiedliche Internetanwendungen und -dienste unterstützt. Dabei werden Funktionen wie Dateitransfer (z. B. File Transfer Protocol, FTP), Electronic Mail (z. B. Simple Mail Transfer Protocol, SMTP), Directory Service (z. B. Lightweight Directory Access Protocol, LDAP) oder World Wide Web (z. B. Hypertext Transfer Protocol, HTTP) bereitgestellt. Ein synonym verwendeter Begriff ist Web-Based Training (WBT).

Zudem existieren verschiedene Begriffe, die einzelne Instanzen von Lernumgebungen charakterisieren. So kennzeichnet der Begriff der *Mobilen Lernumgebung (MLU)* eine Lernumgebung, die durch die Verwendung mobiler Endgeräte wie Personal Digital Assistants (PDA) oder Notebooks mit entsprechender Funkverbindung ortsunabhängig eingesetzt werden können.

Ein zentraler und weit verbreiteter Begriff insbesondere im englischen Sprachraum ist der Begriff *Learning Technologies (Lerntechnologien)*. Obwohl dieser Begriff in der Anwendung häufig einen technologiezentrierten Ansatz impliziert, wird der Begriff in dieser Arbeit in seiner neutralen Bedeutung verwendet.

Lerntechnologien (Learning Technologies, LT) bezeichnen Technologien (Hardware, Software, Netzwerke) und korrespondierende Methoden, die im Kontext des computerunterstützten Lernens verwendet werden.

Lerntechnologien implizieren damit die adäquate Verwendung von Computern im Zusammenhang des computerunterstützten Lernens und die damit zusammenhängenden Inhalte und didaktischen Konzepte. Die jeweils angemessene Planung der verbundenen didaktischen Methoden wird dabei vorausgesetzt. Häufig werden Begriffe und Schlagworte wie *Hypermedia* oder *Multimedia* synonym zu diesem Begriff verwendet, obwohl durch diese Begriffe nur Teilbereiche (siehe Abschnitt 2.2) beschrieben werden. Daher wird im Folgenden der umfassende Begriff der Lerntechnologien verwendet.

Die in dieser Begriffsabgrenzung definierten zentralen Begriffe beschreiben eine Auswahl der vielfältigen Konzepte und Methoden des computerunterstützten Lernens. Weitere Begriffe werden nicht eingeführt, da im Folgenden die Charakterisierung von Lernprozessen nicht durch missverständliche Begriffe, sondern durch eine exakte Beschreibung vorgenommen wird.

2.2 Klassifikation von Lerntechnologien

Wie bereits erwähnt, existiert eine Vielzahl von Konzepten und Methoden des computerunterstützten Lernens und eine noch höhere Anzahl von Instanzen computerunterstützter Lernumgebungen. Um diese Ansätze zu strukturieren, werden im Folgenden verschiedene Klassifikationen des CUL bzw. von Lernumgebungen aufgezeigt.

2.2.1 Funktionale Klassifikation

Diese Klassifikation betrachtet den Funktionsumfang von Systemen, die im Kontext des computerunterstützten Lernens eingesetzt werden. Dabei werden Funktionen zur Unterstützung von Entwicklungs-, Administrations- und Lernprozessen betrachtet. [Bode1990] verwendet beispielsweise eine Klassifikation in *Entwicklungswerkzeuge* für Lernumgebungen sowie *Anwendungswerkzeuge* für Lehrende und Lerner. *Entwicklungswerkzeuge* bezeichnen Software, die im Entwicklungsprozess computerunterstützter Lernumgebungen eingesetzt werden. Dabei können diese spezifisch für den Einsatzbereich des CUL konzipiert sein (Autorensysteme, Entwicklungssprachen). Des Weiteren werden Standardtools wie Textverarbeitung, Datenbanken, Simulationssprachen oder Grafiksysteme eingesetzt. Als zweite Klasse werden *Anwendungswerkzeuge* für Lehrende und Lernende gesehen. Seitens der Lehrenden werden insbesondere Tools zur Prüfungsadministration, Unterrichtsverwaltung und -organisation eingesetzt, während auf Lernerseite spezifische Lernumgebungen verwendet werden (siehe Abschnitt 2.2.2).

Die strikte Trennung dieser Klassifikation ist in den letzten Jahren zunehmend aufgehoben worden. So beziehen Administrationssysteme häufig Entwicklungsfunktionalitäten ein, so dass die beschriebene Trennung nicht aufrechterhalten werden kann. Weiterhin werden zunehmend Schnittstellen zu betriebswirtschaftlichen Informationssystemen (z. B. Control-

ling-Systeme, Personalentwicklungssysteme) geschaffen, so dass Lerntechnologien immer in einem organisationalen Kontext gesehen werden müssen. Lernmanagementsysteme (LMS) sind dabei übergeordnete Systeme, die sämtliche Anwendungs- und Entwicklungswerkzeuge integrieren. Eine Übersicht der Systemklassen ist in Abbildung 2 dargestellt.

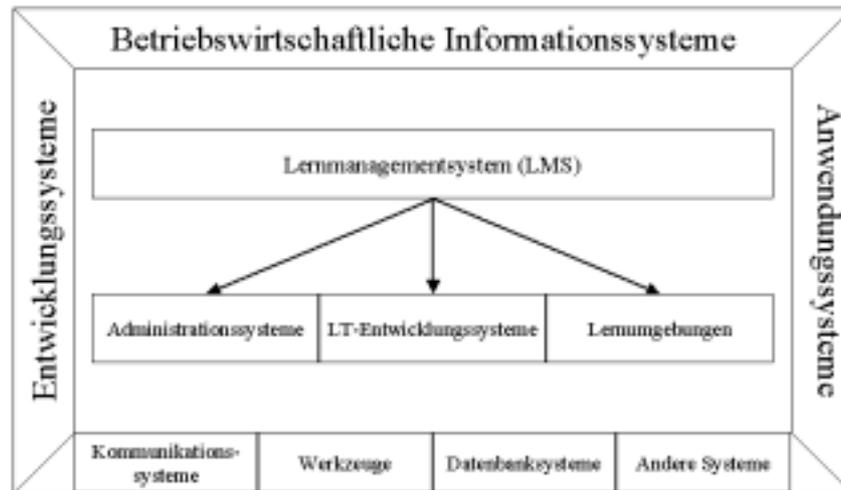


Abbildung 2: Einordnung der LT-Systemklassen

Lernmanagementsysteme (LMS) sind Informationssysteme, die die Durchführung von Entwicklungs- und Lernprozessen ermöglichen. Dabei sind diese Systeme immer Teil eines Gesamtsystems. Die folgenden Systemklassen können als Subsysteme von LMS angesehen werden:

1. *Administrationssysteme* unterstützen die Verwaltung, Steuerung, Kontrolle und Durchführung von Entwicklungs- und Lernprozessen. Dies umfasst zum Beispiel Aufgaben- und Terminverwaltung, Kursmanagement, Prüfungsverwaltung, Tracking und Monitoring.
2. *LT-Entwicklungssysteme* unterstützen Entwicklungsprozesse computerunterstützter Lernumgebungen. Diese Systemklasse umfasst Autorensysteme oder spezifische Entwicklungssprachen.
3. *Lernumgebungen* umfassen alle Informationssysteme, die im Rahmen von Lernprozessen genutzt werden.

Die genannten Klassen greifen zudem auf andere Systeme zu, die in einer Organisation genutzt werden. So werden sowohl in Lern- als auch in Entwicklungs- und Administrationsprozessen Standardwerkzeuge wie Textverarbeitung oder Bildbearbeitungsprogramme verwendet. Kommunikationssysteme sind ebenso Bestandteil betrieblicher Informationssysteme und werden ebenfalls von allen LT-Systemen genutzt. Häufig werden jedoch Lerntechnologien getrennt von betrieblichen Anwendungssystemen gesehen. So ist z. B. häufig eine organisatorische und sachlogische Trennung von Personalentwicklungssystemen und Lerntechnologien zu beobachten, [vgl. AdKP1998a, AdKP1998b, AdKP1999]. Daher müssen Schnittstellen zwischen Lerntechnologien und betrieblichen Informations-

systemen geschaffen werden, die den Datenaustausch und die Integration von Prozessen sicherstellen. Ein Konzept für derartige Schnittstellen wird ausführlicher in Kapitel 4 beschrieben. Die zuvor genannten Klassen können jedoch nicht isoliert betrachtet werden. Es müssen daher genaue Funktionsspezifikationen vorliegen, um den Systemumfang determinieren zu können. So werden in neueren Entwicklungen weitere Funktionen in LMS integriert. [Henr2001] schlägt zum Beispiel die Erweiterung von LMS um folgende Funktionen vor: Annotationsfunktionen, Integration natürlicher Sprache, Multimedia-Interaktion, Qualitätskontrolle, spontane Gruppenbildung, Akkreditierung, innovative strukturierte Interaktionen.

2.2.2 Methodische Klassifikation

Viele Klassifikationen verwenden als Klassifikationskriterium eine Lernmethode oder zugrunde liegende Lerntheorie [Eule1992, Fick1992, Bode1990]. Exemplarisch sei hier die von [Bode1990] verwendete Klassifikationen aufgeführt. Es wird zwischen Hilfesystemen, Lernergesteuerten Systemen, Trainingssystemen, Tutoriellen Systemen, Simulationen/Planspielen und Problemlösungssystemen unterschieden. Diese Klassifikation ist jedoch nicht disjunkt und lässt ferner keine eindeutige Einordnung von Lernumgebungen zu. Die häufige Verwendung dieser Klassifikation ist insbesondere durch die Erklärungsfunktion dieser Ansätze begründet. Durch die intuitive Klassifikation lassen sich sowohl historische Entstehungsansätze als auch Entwicklungstendenzen aufzeigen. Daher wird dieser Klassifikationsansatz als einführende Übersicht aufgegriffen.

2.2.2.1 Hilfesysteme

Hilfesysteme sind ursprünglich als Ergänzung zu Anwendungssystemen entwickelt worden. Hilfesysteme bieten eine Unterstützung von Anwendern in der Regel während der Laufzeit. Der Anwender hat die Möglichkeit, selbstständig die Hilfefunktion über ein integriertes Menü anzufordern. Das System nimmt daraufhin eine Analyse der Problemstellung vor und präsentiert dem Anwender die angeforderte Hilfe. Während Hilfesysteme ursprünglich nur auf eine Hilfeanforderung des Benutzers reagierten, versuchen Weiterentwicklungen dieser Systemklasse Probleme des Benutzers durch die Analyse seines Verhaltens, seines Wissensstandes und aufgrund der durchgeführten Aktivitäten zu identifizieren und daraus folgend aktiv Hilfefunktionen anzubieten. Diese Systeme werden häufig als Intelligente Hilfesysteme bezeichnet.

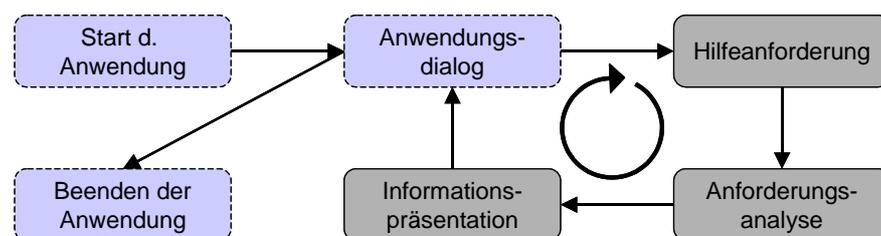


Abbildung 3: Hilfesysteme [Bode1990]

[Gray1998] stellt jedoch fest, dass Hilfesysteme häufig nicht den Anforderungen der Benutzer entsprechen. In vielen Fällen bieten Hilfesysteme nur eine wenig elaborierte Umsetzung der textuellen Systemdokumentation, so dass das Auffinden der gewünschten Information für den Anwender einer eigenen Hilfe bedarf. [Caro1990] kritisiert insbesondere, dass Hilfesysteme häufig die Probleme eines schlechten Designs der Benutzerschnittstellen der Anwendungssysteme verlagern.

[RaRe2001] folgern, dass die Beschränkung von Hilfsfunktionen auf spezifische Anwendungen wesentliche Nachteile gegenüber Systemen haben, die Benutzer bei komplexen, wissensintensiven Arbeitsabläufen unterstützen. *Electronic Performance Support Systems (EPSS)* sollen die Leistungsfähigkeit der Anwender verbessern, indem neben Hilfsfunktionen zusätzliche Funktionalitäten wie Entscheidungsunterstützende Systeme oder Unterstützung zu Arbeitsabläufen angeboten werden. [RaRe2001] sehen langfristig eine Zusammenführung von traditionellen Hilfesystemen, EPSS und Wissensmanagementsystemen, so dass die Systemklasse Hilfesysteme wesentlich über ihre ursprünglichen Funktionalitäten hinausgeht.

2.2.2.2 Lernergesteuerte Systeme

Lernergesteuerte Systeme folgen nach [Bode1990] zwei Grundprinzipien: Lerneinheiten werden durch die Lernumgebung zur Verfügung gestellt, des Weiteren besitzen Lernende selbst die Kontrolle, welche Lerneinheiten zu welchem Zeitpunkt bearbeitet werden. Variationsmöglichkeiten sind die Anpassung des Schwierigkeitsgrades, Zahl und Umfang von Trainingseinheiten oder die lernergesteuerte Auswahl der Lernerfolgsüberprüfung. Die Ähnlichkeit zur Klasse der Hilfesysteme ist offensichtlich, da in beiden Fällen der Lernende Informationen anfordert, die vom System präsentiert werden. Der vorrangige Unterschied ist die Zielsetzung, da lernergesteuerte Systeme auch unabhängig von einem spezifischen Anwendungssystem sein können und damit aus einem anderen Entwicklungsansatz heraus entstanden sind.

Zur Informationspräsentation und -repräsentation werden gerade durch die zunehmende Verbreitung des World Wide Web (WWW) häufig Hypertext-, Hypermedia- und Multimediaanwendungen verwendet. Unter *Hypertext* versteht man nichtlineare Texte, die durch Knoten und Verbindungen (Links) in Beziehung gesetzt werden [vgl. Blum1998]. Das Konzept Multimedia wird sowohl als Definition als auch in seiner Anwendung in der Literatur umfangreich diskutiert, eine Übersicht findet sich bei [Blum1998] und [IsKl1997]. [Blum1998] bezeichnet *Multimedia* als digitalisierte, multimodale, multicode Repräsentationsform von Informationen, die durch den Computer integriert wird und durch Anwender interaktiv nutzbar ist. *Hypermedia* ist daraus folgend nichtlinear strukturierte, multimediale Information, die interaktiv durch Anwender nutzbar ist. Es ist anzumerken, dass die Begriffe Hypermedia und Multimedia häufig synonym verwendet werden.

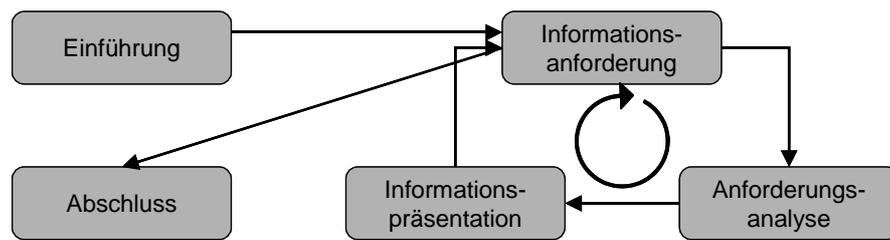


Abbildung 4: Lernergesteuerte Systeme [Bode1990]

Der Einsatz multimedialer und hypermedialer Methoden zur Verbesserung von Lernprozessen ist eine zentrale Fragestellung im Forschungsfeld des computerunterstützten Lernens. Gerade in Lernergesteuerten Systemen ist die didaktische Planung des Multimediaeinsatzes ein kritischer Erfolgsfaktor.

Durch die Erweiterung der Lernergesteuerten Systeme um multimediale Inhalte ändert sich auch der Umfang dieser Systemklasse. So werden zum Beispiel klassische Vorlesungen oder Seminare per Audio- und Videoaufzeichnung Lernenden bei Bedarf zur Verfügung gestellt (Lecture on Demand). Die Vielfalt der Ausgestaltung lernergesteuerter Systeme bedarf daher einer weiteren Detaillierung, um Aussagen über mögliche Einsatzbereiche machen zu können, was jedoch den Rahmen der vorliegenden Arbeit bei Weitem übersteigen würde.

2.2.2.3 Trainingssysteme

Trainingssysteme basieren in ihrer Ursprungsform auf der Theorie der *Programmierten Instruktion*. Die programmierte Instruktion ist eine Methode zur Umsetzung der behavioristischen Lerntheorie nach [Skin1954, Skin1958] zur Entwicklung von Lernumgebungen. Derartige Systeme werden auch als *Drill & Practice-Systeme* (siehe auch 2.3.4.1) bezeichnet. Sie basieren auf der Aufteilung von Lerninhalten in handhabbare Einheiten und der Operationalisierung von Lernzielen, die sukzessive präsentiert und anschließend durch laufende Übung vertieft werden. Trainingssysteme setzen einen Schwerpunkt auf die Einübung und Vertiefung bereits vorhandenen Wissens. Der traditionelle Begriff des *Computer Based Training (CBT)* fasst diese Klasse von Systemen zusammen.

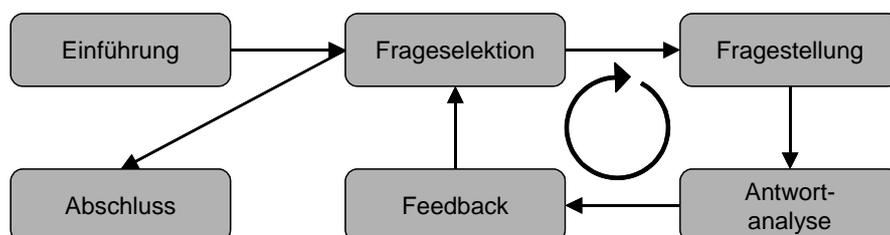


Abbildung 5: Trainingssysteme [Bode1990]

Kritik an diesen Ansätzen bezieht sich insbesondere auf die mangelnde Interaktion, die fehlende Einbindung in den Kontext des Lernenden sowie mangelnde Konstruktion von Wissen durch den Lernenden selbst (vgl. 2.3.4.3). Dennoch basieren auch heute eine Vielzahl von Lernumgebungen auf diesem Ansatz und werden insbesondere bei relativ eindeutigen, stabilen Lerninhalten und einer großen Anzahl von Nutzern verwendet.

Neuere Ansätze von Trainingssystemen versuchen, positive Lernerfolge und zusätzliche Motivation durch Einbeziehung des Kontexts der Lernenden zu erreichen. Im Rahmen des *situierten Lernens* [Kerr2001] werden für den Lernenden bekannte Situationen dargestellt, so dass die zu übenden Inhalte in einer entsprechenden Anwendungssituation verwendet werden können. Auch hier ist somit eine Überschneidung zur Klasse der Simulationssysteme erkennbar.

2.2.2.4 Tutorielle Systeme

Tutorielle Systeme übernehmen neben der Präsentation der Lerninhalte auch die Führung der Lernenden durch die Inhalte [Bode1990]. Tutorielle Systeme ermöglichen den Lernenden, vordefinierte Lernwege durch die Lerninhalte zu durchlaufen. Hinzu kommen Hilfefunktionen, die antizipierte Probleme der Lernenden lösen sollen. Das System tritt also an die Stelle des Lehrenden. Eine Erweiterung dieser Ursprungsform sind *Intelligente Tutorielle Systeme (ITS)*. Sie bezeichnen Systeme, die Wissen entsprechend der jeweiligen Benutzerbedürfnisse vermitteln. Lerninhalte, Lernmethoden oder Präsentationsformen werden aufgrund verschiedener Charakteristika der Lernenden (z. B. Lernpräferenzen) individuell angepasst. Da die Anpassung häufig im Vordergrund steht, werden diese Systeme auch als *Adaptive Lernumgebungen* bezeichnet. Anstelle des Lehrenden steht eine tutorielle Systemfunktion zur Beantwortung von Fragen oder zur Überprüfung zur Verfügung.

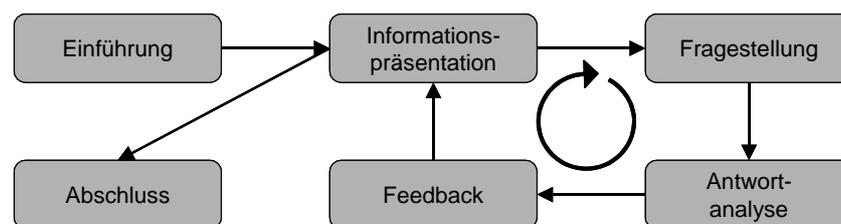


Abbildung 6: Tutorielle Systeme [Bode1990]

Es stellt sich die Frage, inwieweit ein solches System die Funktion eines Lehrenden bzw. einer persönlichen Betreuung übernehmen kann. Die individuelle Anpassung an Lernpräferenzen ist als wichtiger anzusehen als die vollständige Ersetzung menschlicher Betreuer. Gerade durch Weiterentwicklungen im Bereich der Benutzerschnittstellen und der Mensch-Maschine-Interaktionen werden Lernumgebungen so gestaltet, dass sie intuitiv bedienbar werden [vgl. KiPa1997, PRKO1998, KiPR2001, Oppe2001].

2.2.2.5 Problemlösungssysteme

Problemlösungssysteme werden teilweise den Tutoriellen Systemen zugeordnet, da die Konzeption ähnlich ist. Problemlösungssysteme konfrontieren Lernende mit einer konkreten Problemstellung (z. B. einer Fallstudie), die mit einer Hilfestellung durch das System von den Lernenden gelöst werden muss. Einerseits werden hier Lerninhalte angewendet, andererseits wird das Erkennen und Lösen von Problemstellungen erlernt.

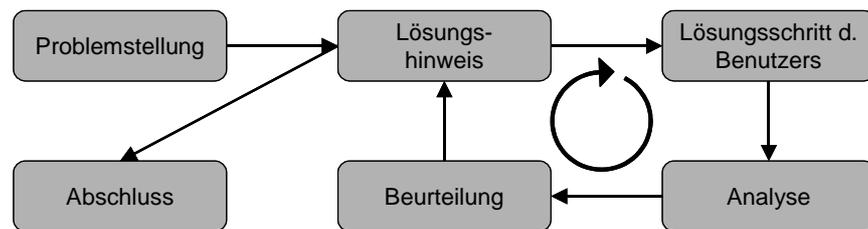


Abbildung 7: Problemlösungssysteme [Bode1990]

2.2.2.6 Simulations- und Spielsysteme

Simulationssysteme ermöglichen es dem Lernenden, reale Konzepte anhand eines computerunterstützten Modells zu erlernen. Innerhalb eines Szenarios, das eine reale Situation abbildet, können Lernende Aktionen zur Modellbeeinflussung vornehmen und die Wirkung der Aktion nachvollziehen, analysieren und evaluieren. Ein Schwerpunkt liegt auf der Sammlung von Erfahrungen, die nach [Ruoh1995] realen Erfahrungen ähneln und positive Auswirkungen auf den Lernerfolg haben. *Spielsysteme* haben als zusätzliches Element einen Wettbewerbscharakter. Mehrere Lernende arbeiten in einer gemeinsamen Lernumgebung; der Wettbewerbscharakter dient als Anreiz, um die Motivation zu erhöhen.

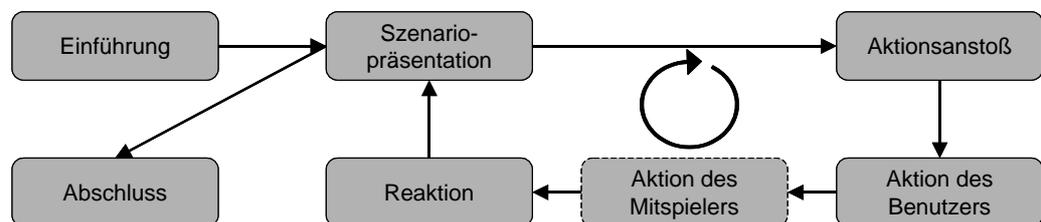


Abbildung 8: Simulations-/Spielsysteme [Bode1990]

Ein weiterer Begriff, der häufig als Schlagwort für Spielsysteme verwendet wird, ist *Edu-tainment*. Dieser Begriff impliziert den Unterhaltungscharakter und versucht, die Motivationalelemente aus Computerspielen mit Lernumgebungen zu verknüpfen, wobei hier der Wettbewerbscharakter in den Hintergrund tritt.

2.2.2.7 Kollaborative Lernumgebungen

Zusätzlich zu den in der Klassifikation von [Bode1990] identifizierten Klassen werden an dieser Stelle kollaborative Lernumgebungen aufgeführt. Kollaborative Elemente können zwar in allen der oben genannten Klassen vorkommen, doch die zunehmende Bedeutung des Gruppenlernens (insbesondere mithilfe des WWW) rechtfertigt eine gesonderte Auf-führung.

[BiKö2001] definieren Kollaboration als „[...] Philosophie der Interaktion vor dem Hinter-grund einer gemeinsamen strategischen Zielsetzung.“ In kollaborativen Lernumgebungen ist die primäre Zielsetzung das gemeinsame Lernen spezifischer Lerninhalte; ergänzt um weitere Ziele, wie etwa der Erwerb sozialer Kompetenzen (z. B. Teamarbeit, -fähigkeit). Kollaborative Lernumgebungen stellen Werkzeuge zur Verfügung, die die Kommunikation zwischen Lernenden ermöglichen und eine gemeinsame Arbeitsoberfläche bieten (z. B. gemeinsame Dokumentenbearbeitung).

2.2.2.8 Kognitive Werkzeuge

Kognitive Werkzeuge unterstützen den Wissenserwerb nicht durch die direkte Präsentation von Lerninhalten oder Problemstellungen, sondern stellen vielmehr Werkzeuge zur Verfü-gung, um kognitive Konzepte darzustellen, zu abstrahieren und selbst zu generieren. [Schu1996] betont die Vorteile dieser Systemklasse: „Kognitive Werkzeuge [...] erlauben einen ganzheitlichen Zugang zum Lernen, d.h. sie gestatten es, sich den größeren Konzep-ten allmählich durch entdeckendes Verhalten zu nähern, Konzepte zu generieren, und erst im Prozess einzelne Subkonzepte zu lernen, während Instruktionssysteme einen analyti-schen Zugang favorisieren.“ Als Beispiel seien an dieser Stelle kognitive Landkarten ge-nannt. Kognitive Landkarten visualisieren Konzepte und deren Zusammenhänge. Konzep-te, Strukturen und ihre Relationen werden zunächst mithilfe eines graphischen Werkzeuges repräsentiert. Die Bearbeitung und das Erlernen erfolgt nicht durch die Manipulation der Konzepte selbst, sondern durch die Manipulation ihrer graphischen Repräsentation. Diese Assoziation soll durch ihre Ähnlichkeit zu mentalen Strukturen den Lernprozess fördern.

2.2.2.9 Kritische Würdigung

Wie bereits erwähnt bietet die methodische Klassifikation nur einen Überblick über histo-rische Entwicklungsansätze und prinzipielle Trends des computerunterstützten Lernens. Es wurde bereits gezeigt, dass es vielfältige Überschneidungen der einzelnen Systemklassen gibt. [Collis2001] beschreibt *Konvergenz* als einen wichtigen Trend von Konzepten, Me-thoden und Applikationen des computerunterstützten Lernens. Dabei werden verschiedene Konzepte, Methoden und Applikationen mit unterschiedlichen Zielsetzungen, Anwen-dungsgebieten und Umsetzungen zusammengeführt. Daher ist es nicht mehr möglich, heu-tige Systeme eindeutig in eine der oben genannten Klassen einzuordnen. Des Weiteren ist es nicht möglich, für derartige Systemklassen Aussagen über den Nutzen zu treffen und

damit eine Entscheidungsgrundlage zu liefern. Vielmehr ist eine detaillierte Beschreibung der Lernsituation, des Kontextes und der Lernumgebung notwendig, um transferierbare Schlüsse über die Verwendbarkeit einer Lernumgebung ziehen zu können.

2.3 Beschreibungsmerkmale von Lernumgebungen

Mehrdimensionale Klassifikationen versuchen, Lernumgebungen aufgrund verschiedener relevanter Dimensionen zu klassifizieren. Ziel derartiger Klassifikationen ist die Identifikation relevanter Kriterien, die eine genaue Charakterisierung bieten und als Entscheidungsgrundlage dienen können. Zentral ist die Fragestellung, welche Art von Lernumgebungen in bestimmten Lernsituationen mit spezifischen Randbedingungen erfolgversprechend ist. Zu diesem Zweck werden im Folgenden Merkmale erläutert, mit denen Lernumgebungen detailliert beschrieben werden können.

Es existiert derzeit keine Klassifikation, die Lernumgebungen eindeutig und vollständig charakterisiert. Je nach Fachgebiet, didaktischem Ansatz oder aufgrund persönlicher Präferenzen unterscheiden sich Beschreibungs- und Entscheidungskriterien wesentlich. Die Problematik, Lernumgebungen oder auch einzelne Lernobjekte mithilfe einer minimalen Anzahl von Merkmalen zu beschreiben, wird im Rahmen der Diskussion bez. Metadaten (Abschnitt 4.2) ausführlicher aufgegriffen. In diesem Abschnitt werden zunächst verschiedene Modelle erläutert und Merkmale abgeleitet, die zur heuristischen Beschreibung von Lernprozessen verwendet werden.

Lernprozesse werden nicht ausschließlich durch die benötigte Software charakterisiert; es ist unerlässlich, die Rahmenbedingungen detailliert zu spezifizieren. Das Berliner Modell [HeSO1965] ist ein lehrtheoretischer Ansatz, um Lernprozesse zu beschreiben. Es wurde zur Analyse und Gestaltung des Unterrichts entwickelt und unterscheidet Bedingungsfelder und Entscheidungsfelder. Während die Bedingungsfelder die Lernsituation und den Kontext beschreiben, betreffen Entscheidungsfelder die Gestaltung des Unterrichts.

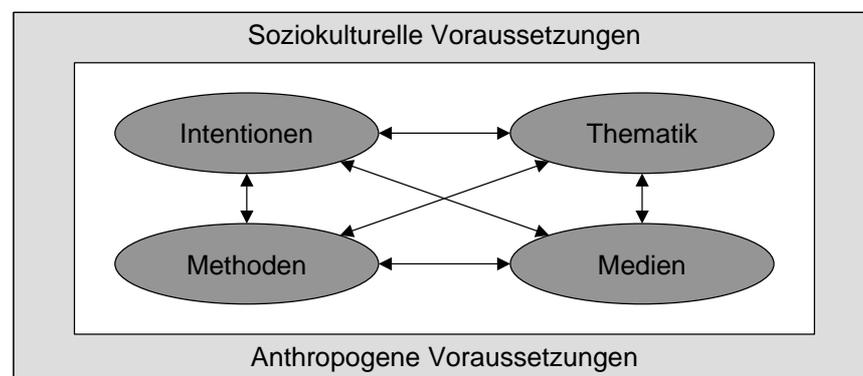


Abbildung 9: Berliner Modell [HeSO1965]

Soziokulturelle Voraussetzungen betreffen den Kontext, in dem der Unterricht geplant wird. Dieser umfasst das Bildungssystem oder allgemeine Trends (z. B. Entwicklung zur Informationsgesellschaft). *Anthropogene Voraussetzungen* beschreiben die Akteure. So werden die Eigenschaften und Unterschiede von Lehrenden und Schülern (z. B. Geschlecht, Motivation, Vorwissen) erfasst.

Intentionen beschreiben die Planung des Unterrichts aus Sicht des Lehrenden. Es wird geplant, welche pädagogischen Absichten und Lernziele für den Unterricht in Betracht kommen. Die *Thematik* erfasst inhaltliche Schwerpunkte des Unterrichts. Ferner werden *Methoden* zur didaktischen Konzeption des Unterrichts und einzusetzende *Medien* geplant.

Dieses Modell ist generell nur zur Einordnung des didaktischen Kontextes geeignet. Somit eignet es sich nur als grobe Planungshilfe, da der Einsatz von Lerntechnologien in diesem Modell nur unzureichend spezifiziert werden kann. Daher müssen zusätzliche Beschreibungsmerkmale eingefügt werden.

[Kerr2001] unterscheidet vier Kategorien zur Beschreibung von Lernumgebungen:

Die Kategorie *Zielgruppe* charakterisiert eine Gruppe von Lernenden, die potenzielle Teilnehmer einer Lernumgebung sind. Es wird nach soziodemographischen Merkmalen (Gruppengröße, geographische Verteilung, Alter, Geschlecht, Schulabschluss, IT-Kenntnisse, Kaufbereitschaft), Vorwissen, Lernmotivation, Lerngewohnheiten, Lerndauer, Einstellungen/Erfahrungen sowie Lernort/Medienzugang unterschieden. Die Zielgruppe umfasst Lernende, nicht aber Lehrende oder weitere am Lernprozess Beteiligte. Aufgrund der starken Interdependenzen von Lehrenden und Lernenden müssen jedoch weitere Akteure in dieser Kategorie berücksichtigt werden. Ein weiterer Grund für die Erweiterung ist die Verwendung zugekaufter Lernumgebungen, die nicht selbst konzipiert und implementiert wurden: Anforderungen, die an den Lehrenden bzw. Betreuer gestellt werden, müssen in einer detaillierten Anforderungsdefinition erfasst werden.

Ziele/Inhalte beschreiben die Zielsetzung und inhaltliche Ausgestaltung von Lernumgebungen. Dabei wird zwischen Zielen (Projektzielen, Lehrzielen, Lernzielen, Kommunikationszielen) und Inhalten unterschieden. Die inhaltliche Komponente umfasst dabei neben der Wissensrepräsentation auch die Beschreibung des Lernangebots (Aktionsformen, Visualisierung, Meta-Aktionen, Einbettung in Handlungswelten).

Die *Didaktische Struktur* beschreibt die mediendidaktische Konzeption einer Lernumgebung. Es werden dabei vier Strukturprinzipien unterschieden: Exposition, Exploration, Konstruktion und Kommunikation. Diese Kategorie umfasst somit einerseits didaktische Methoden, einschließlich Lernerfolgskontrollen und Feedback, andererseits werden Kommunikationswege impliziert.

Die *Lernorganisation* betrachtet die räumliche und zeitliche Organisation von Lernumgebungen. [Kerr2001] betont, dass diese Dimension als eigenes Entscheidungsfeld betrachtet werden muss, nicht nur als Randbedingung. Das Lernangebot wird somit nicht nur auf die Konzeption einer Software beschränkt, sondern um die Schaffung von Lernsituationen und deren Einsatz erweitert.

Die beschriebene Klassifikation ermöglicht eine vollständige und verständliche Einordnung von Lernumgebungen. Dennoch bietet sie keine eindeutigen Anhaltspunkte für den benötigten Umfang einer Lernumgebungsspezifikation. Im Folgenden werden die notwen-

digen Kategorien zur Beschreibung von Lernumgebungen diskutiert. Daraus resultiert ein Schema, das eine genaue Beschreibung von Lernumgebungen ermöglicht.

2.3.1 Kontext

Der Kontext beschreibt das potenzielle Anwendungsgebiet einer Lernumgebung und bildet Einsatzmöglichkeiten und etwaige Restriktionen ab. Diese Abgrenzung ist aufgrund der bereits erwähnten Vielfalt möglicher Einsatzgebiete unerlässlich. Des Weiteren unterscheiden sich Lernumgebungen, wie in Abschnitt 2.2.2 exemplarisch aufgeführt, bezüglich ihrer didaktischen Konzeption. Die Konzeption von Lernumgebungen kann einerseits sehr umfassend sein (z. B. Planung von Personalentwicklungsprogrammen), andererseits kann nur die Planung einer einzelnen Lerneinheit betrachtet werden. Eine mögliche Beschreibung des didaktischen Kontextes wird durch die Ebenen des didaktischen Handelns nach [FIHa1977] vorgenommen. Ferner ist das intendierte Einsatzgebiet der konzipierten Lernumgebungen abzugrenzen. Dabei müssen Informationen über die Organisation verfügbar sein, für die eine Lernumgebung konzipiert wurde. Schließlich muss die Einordnung von Lernumgebungen in den organisationalen Kontext spezifiziert werden.

2.3.1.1 Ebenen des didaktischen Handelns

Eine mögliche Abgrenzung der Inhalte und Aufgaben einer Lernumgebung bieten die Ebenen des didaktischen Handelns. *Didaktisches Handeln* bezeichnet die „[...] Organisation von Lernprozessen menschlicher Individuen.“ [FIHa1977]. *Didaktische Ebenen* dienen dazu, diese Handlungen hinsichtlich ihrer Inhalte und Ziele voneinander abgrenzen zu können. Unterschieden werden die Ebenen A bis E (Abbildung 10).

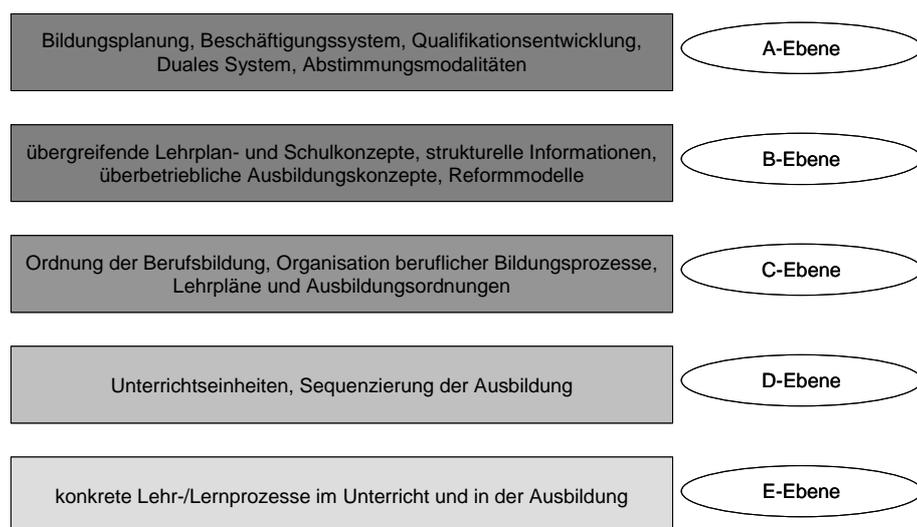


Abbildung 10: Ebenen des didaktischen Handelns

Dabei versteht man unter der *A-Ebene* die „Gestaltung der institutionellen, personellen und konzeptionellen Rahmenbedingungen“ [FIHa1977]. Folgende Inhalte werden abgedeckt: Grundlegende Gestaltung der Berufsausbildung (z. B. duales System), Einbeziehung von Eltern, Schülern, aber auch Gemeinden, Verbänden, Unternehmen (z. B. Einrichtung von Hochschulen durch Großunternehmen), Entscheidungen über den prozentualen Anteil der Bildungsausgaben am Bruttosozialprodukt und die zu berücksichtigenden Lerninhalte sowie Formulierung von Bedürfnissen der Gesellschaft, denen das Bildungswesen Rechnung zu tragen hat.

Die *B-Ebene* bezieht sich auf die „Gestaltung übergreifender Lehrplan- und Schulkonzepte“ [FIHa1977]. Dies betrifft die Entscheidung über die Zielrichtung, mit der die auf der A-Ebene verabschiedeten Inhalte vermittelt werden sollen sowie über Grundsätze der Lernorganisation.

In der *C-Ebene* werden Aktivitäten zusammengefasst, die auf die „Gestaltung von Lernbereichen und Unterrichtskonzeptionen“ [FIHa1977] abzielen. „Im besonderen geht es darum, für einzelne Bildungsstufen, Fächer, Schultypen oder Gruppen von Lernenden Pläne auszuarbeiten, zu verwirklichen und die Ergebnisse auszuwerten“ [FIHa1977].

In der *D-Ebene* werden Maßnahmen der „Gestaltung von Unterrichtseinheiten“ [FIHa1977] gebündelt. Dabei sind unter Unterrichtseinheiten nicht einzelne Lektionen oder Stunden zu verstehen, sondern lektionsübergreifende Planungseinheiten.

Die *E - Ebene* umfasst die „Gestaltung von Lehr-/Lernsituationen“ [FIHa1977]. Dabei werden hauptsächlich Verhaltenstrainings von Lehrenden betrachtet, die z. B. Techniken der Fragestellung, Erklärung, Reaktion auf Aggressionen und des Lobens behandeln.

Diese Abgrenzung ermöglicht eine Einordnung der Zielsetzung von Lernumgebungen und Entwicklungsumgebungen. Kleinere Lernumgebungen umfassen häufig nur einzelne Unterrichtseinheiten und sind damit auf der E-Ebene angesiedelt. Umfassende Planungen für Lernumgebungen sind dagegen auf der C-Ebene einzuordnen. Dabei werden nicht nur einzelne Inhalte betrachtet, vielmehr werden organisatorische Maßnahmen und weitere Planungsmechanismen (z. B. Curriculumentwicklung) einbezogen.

2.3.1.2 Organisationale Voraussetzungen

Organisationale Voraussetzungen beschreiben die Einordnung einer Lernumgebung innerhalb einer Organisation (Unternehmen, Hochschule, Schule). Diese Einordnung kann auf inhaltlicher Ebene erfolgen, d. h., dass eine Lernumgebung innerhalb eines Curriculums oder innerhalb von Personalentwicklungsplänen zugeordnet wird. Ebenso werden auch organisatorische Voraussetzungen erfasst. Beispielsweise können vertragliche Regelungen oder sonstige Vereinbarungen einen bestimmten Zeitanteil für Weiterbildungsmaßnahmen gewährleisten.

Des Weiteren kann eine Zuordnung von Lern- und Geschäftsprozessen erfolgen. Existiert in einer Organisation ein Geschäftsprozessmodell, so sollten Lernprozesse und damit verbundene Kompetenzen den jeweils entsprechenden Geschäftsprozessen zugeordnet werden [vgl. AdKP1998a, AdKP1998b, AdKP1999].

2.3.2 Lernsituation

Die Lernsituation gibt eine Übersicht über die konkrete Anwendung einer Lernumgebung [vgl. Kerr2001]. In der Praxis ist es oft zu beobachten, dass Lernumgebungen außerhalb des konzipierten Kontextes verwendet werden und der Erfolg somit gefährdet ist. Daher ist die Lernsituation, für die eine Lernumgebung geeignet ist, zu spezifizieren. Die Lernsituation bezieht sich hier insbesondere auf die räumliche und zeitliche Einordnung. Weitere Faktoren wie Methoden oder Aktoren werden als eigenes Merkmal spezifiziert.

2.3.2.1 Zeitliche Einordnung

Die zeitliche Einordnung bezieht sich auf die beiden Faktoren Lernzeitpunkt und Lerndauer.

Der *Lernzeitpunkt* gibt an, zu welchen Zeitpunkten die Lernumgebung verwendet wird. Synchroner Lernformen bedeuten, dass die Lerngruppe gleichzeitig am Lernprozess teilnimmt, während bei asynchronen Lernformen die Lerngruppe zu unterschiedlichen Zeitpunkten arbeitet. Häufig werden synchrone und asynchrone Phasen zu Mischformen kombiniert.

Als zweiter Faktor ist die *Lerndauer* anzugeben. Die Dauer geschlossener Lernprozesse ist festgelegt, d. h., dass ein Zeitraum für die Bearbeitung vorgegeben und bekannt ist. Offene Lernprozesse haben dagegen keinen festgelegten Anfangs- und Endzeitpunkt. Diese Form ist gerade bei freiwilligen und selbst initiierten Lernprozessen üblich.

2.3.2.2 Räumliche Einordnung

Die räumliche Einordnung bezieht sich auf die Verteilung der Lernenden und der Lehrenden. In der gerade im Hochschulbereich typischen Präsenzform sind Lehrende und Lernende stets am gleichen Ort. In der Form des Fernstudiums sind Lehrende und Lernende während eines Großteils des Lernprozesses räumlich verteilt. Auch hier sind Mischformen möglich.

2.3.3 Lernziele und Lerninhalte

Die Spezifikation und Abbildung der Lernziele und der damit verbundenen Inhalte ist einer der zentralen Erfolgsfaktoren von Lernumgebungen. Durch die Spezifikation von Lernzielen wird beschrieben, was der Lernende nach Bearbeitung von Lernumgebungen beherrschen soll. Die Inhalte beschreiben die zu erlernenden Konzepte und Methoden.

2.3.3.1 Lernziele

Unter einem *Lernziel* versteht man die „[...] Beschreibung von Eigenschaften, die der Lernende nach erfolgreicher Lernerfahrung erworben hat. Es ist die Beschreibung eines Kataloges von Verhaltensweisen, die der Lernende äußern können soll.“ [Mage1962]. Diese Definition ist sinnvoll, weil sie die Einbeziehung von Methoden zur Erreichung ausklammert. Vielmehr dienen Lernziele als „[...] Grundlage für die Auswahl geeigneten Materials und angemessener Inhalte und Unterrichtsmethoden.“ [Mage1962]. Darüber hinaus werden Lernziele „[...] immer wieder zur Vergewisserung über den erreichten Stand des Lernprozesses“ [Riek1982], also als Grundlage der Lernerfolgskontrolle herangezogen.

Lernziele können nach verschiedenen Kriterien klassifiziert werden. In dieser Arbeit wird nach ihrem Abstraktionsniveau sowie nach der Art ihres Inhalts unterschieden. Beide Klassifikationen sind für die Curriculumentwicklung von Bedeutung, da im Verlaufe der Entwicklung die Ziele für alle Lerninhalte sukzessive präzisiert werden.

2.3.3.1.1 Lernzielklassifikation nach Abstraktionsniveau

In der Literatur werden Lernziele unter anderem nach dem Grad ihres Abstraktionsniveaus geordnet. Man unterscheidet zwischen allgemeinen und konkreten Zielsetzungen. „Der Unterschied [...] darf nun aber nicht so interpretiert werden, als ob die Lernziele unter dem Gesichtspunkt der Ausdrucksform dichotomisiert werden dürften. In Wirklichkeit ist es gar nicht möglich, einen scharfen Trennstrich zwischen allgemeinen und konkreten Zielen zu ziehen, es handelt sich vielmehr um ein Kontinuum“ [Cort1974]. Dieses sogenannte *Abstraktionskontinuum* (Abbildung 11) reicht von sehr allgemeinen Lernzielen bis zu sehr konkreten Lernzielen. Dabei „[...] erscheint es sinnvoll, drei verschiedene Lernzielarten aus dem Abstraktionskontinuum herauszuheben: Lernziele eines sehr hohen Abstraktionsniveaus oder Abstraktionsniveaus 3, eines mittleren Abstraktionsniveaus oder Abstraktionsniveaus 2 und Lernziele eines sehr niedrigen Abstraktionsniveaus oder Abstraktionsniveaus 1.“ [Möll1973].

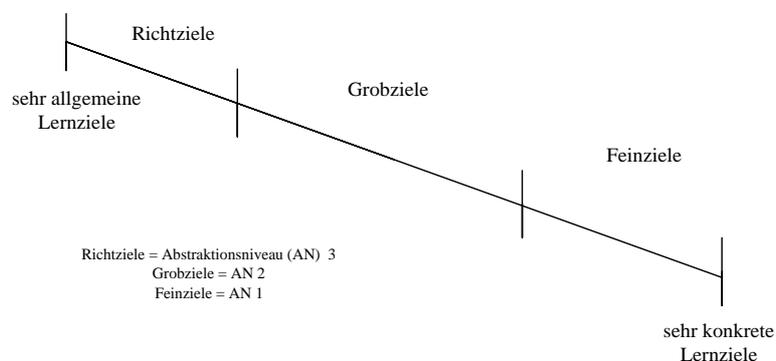


Abbildung 11: Lernzielkontinuum

Diese Einteilung ist nicht unumstritten. Andere Autoren unterscheiden beispielsweise nach impliziten und expliziten Lernzielen, wobei explizite formulierte, implizite dagegen nicht-formulierte Lernziele beschreiben. Weiterhin wird nach materialen und formalen Lernzielen differenziert. Dabei sind materiale Lernziele an ein Objekt gebunden, formale dagegen an ein Subjekt. Schließlich wird auch unterschieden nach unmittelbaren und letzten Lernzielen. Dabei stellen die letzten Ziele Endresultate eines Unterrichts dar, während unmittelbare Ziele die nötigen Zwischenschritte beschreiben [vgl. Cort1974].

Für die Beschreibung von Lernumgebungen wird in dieser Arbeit die Klassifikation nach Abstraktionsniveau ausgewählt, da sie dem Prinzip der inhaltlich-substanziellen Verknüpfung der Planung entspricht, wobei der Output der strategischen Planung als Input in die taktische oder operative Planung eingeht [vgl. PfSt1997]. Insbesondere bei einem evolutionären Planungsprozess ist diese Klassifikation sinnvoll einsetzbar.

2.3.3.1.2 Lernzielklassifikation nach Lerndimensionen

Eine weitere Möglichkeit der Klassifikation stellt die Ordnung nach den Inhalten der Lernziele dar. Dabei unterscheidet man traditionell nach kognitiven, affektiven und psychomotorischen *Lerndimensionen*.

Die *kognitive Dimension* „[...] schließt dabei solche Lernziele ein, die Erinnern oder Erkenntnis von Wissen und die Entwicklung intellektueller Fähigkeiten und Fertigkeiten behandelt.“ [Bloo1956, zitiert nach Bloo1973]. Dies umfasst neben reinem Faktenwissen auch das Verständnis und die Möglichkeit der Anwendung dieses Wissens zu Analyse, Synthese und Evaluation [vgl. Bloo1973].

Die *affektive Lerndimension* beschreibt „Lernziele, die ein Gefühl, eine Emotion oder ein bestimmtes Maß an Zuneigung oder Abneigung betonen. Affektive Lernziele reichen von der einfachen Beachtung bestimmter Phänomene bis zu komplexen, aber in sich konsistenten Qualitäten des Charakters und des Bewusstseins.“ [KrBM1964, zitiert nach KrBM1975]. Dabei werden folgende Abstufungen vorgenommen: Auf etwas aufmerksam werden, Reagieren, Werten, Organisieren der Werte (Wertordnung), Bestimmtheit durch Werte [vgl. KrBM1975]. Insbesondere fallen in diese Kategorie auch Änderungen von Interessenslagen, die Bereitschaft, etwas zu tun und dauerhafte Werthaltungen [vgl. Ja-Me1994].

Unter *psychomotorischen Lernzielen* versteht man schließlich „Lernziele, die Wert legen auf muskuläre oder motorische Fähigkeiten, auf den Umgang mit Material oder Gegenständen oder auf eine Handlung, die neuromuskuläre Koordination erfordern“ [KrBM1975]. Dabei unterscheidet man die Stufen Imitation, Manipulation/Fertigkeiten, Präzision, Handlungsgliederung und Naturalisierung [Diep1999].

Die aufgeführten Abstufungen innerhalb der Lernziele erfolgen dabei nach deren Komplexitätsgraden, auf die bei der Ordnung der Lernziele unter 2.3.4 zurückgegriffen wird. Eine andere Klassifikation, die eher dem Bereich der wirtschaftswissenschaftlichen Forschung zuzuordnen ist, unterscheidet Kenntnisse, Fähigkeiten und Fertigkeiten [Stae1994]. Diese werden auch unter dem Begriff der *Qualifikationen* zusammengefasst. Auch diese Nomenklatur ist grundsätzlich zur Klassifikation von Lernzielen geeignet. Da aber für das didakti-

sche Klassifikationsschema eine detailliertere und weit verbreitete Taxonomie vorliegt, wird dieses für das weitere Vorgehen gewählt.

Neben den drei aufgeführten Dimensionen existieren verschiedene Erweiterungen, von denen hier exemplarisch drei genannt werden. Dies ist zum ersten der Begriff *Schlüsselqualifikation*, worunter (im Gegensatz zu fachlichen Lernzielen) allgemeine Lernziele, wie unter anderem Kooperations-, Kommunikations-, Team-, Problemlösungsfähigkeiten sowie analytisches Denken und Kreativität verstanden werden [Hent1994]. Ähnlich ist der Begriff *extrafunktionale Qualifikation* zu verstehen, der sich „[...] auf normative Orientierungen, wie etwa Verantwortungsbereitschaft, Arbeitsdisziplin, Anpassungsbereitschaft, Flexibilität, Identifikation mit den jeweiligen Organisationszielen“ [Stae1994] bezieht.

Schließlich spricht man in der Didaktik auch von *sozialen Lerninhalten*, die sich auf soziales Verhalten beziehen [Diep1999]. Dabei sind Inhalte wie Teamfähigkeit und soziale Kompetenz von besonderer Bedeutung.

Um eine einheitliche Terminologie zu gewährleisten, wird in dieser Arbeit zwischen kognitiven, affektiven und psychomotorische sowie sozialen Lernzielen unterschieden.

2.3.3.2 Lerninhalte

Lerninhalte sind diejenigen Konzepte, die mithilfe einer Lernumgebung erworben werden sollen. In diesem Abschnitt wird erläutert, wie Lerninhalte beschrieben, repräsentiert und klassifiziert werden können. Lerninhalte sollen so charakterisiert werden, dass daraus Folgerungen für Designentscheidungen getroffen werden können. Zum Beispiel sind bestimmte didaktische Methoden nur für spezifische Lerninhalte sinnvoll nutzbar.

2.3.3.2.1 Domänenwissen

Eine grundlegende Fragestellung bei der Klassifikation des CUL ist die nach den Lerninhalten, die in einer Lernumgebung verwendet werden. Dies bezieht sich primär auf Domänenwissen, das heißt, die Einordnung des Wissens in den Kontext einer spezifischen Domäne. Um Inhalte abgrenzen und in einen Zusammenhang bringen zu können, werden verschiedene Möglichkeiten dargestellt, die meist mit einer bestimmten Repräsentationsform korrelieren. An dieser Stelle wird die Fragestellung der geeigneten Repräsentationsform nicht diskutiert. Dies geschieht vor dem Hintergrund, dass das im weiteren Verlauf der Arbeit entwickelte Essener-Lern-Modell den Autoren die Wahl der Repräsentationsform überlässt. So können zum Beispiel Taxonomien zur Wissensrepräsentation verwendet werden. Taxonomien beschreiben die Strukturen von Wissen und entsprechende hierarchische Zusammenhänge. Die Verwendung von Taxonomien wird im Abschnitt 4.4.3.1 ausführlicher diskutiert.

Eine weitere Möglichkeit ist die Repräsentation durch Ontologien. Unter einer Ontologie versteht man eine explizite und eindeutige Darstellung von Konzepten und Relationen [vgl. MiS1996]. Im Zusammenhang der Entwicklung wissensbasierter Systeme wird unter einer Ontologie das Vokabular und die Konzepte verstanden, die zur Entwicklung künstlicher

Systeme notwendig sind [Mizo1993]. Im Unterschied zur hierarchischen Struktur von Taxonomien werden in Ontologien Konzepte durch Relationen verknüpft, die nicht auf hierarchische Beziehungen beschränkt sind. Weiterhin ist die Möglichkeit gegeben, verschiedene Perspektiven zu konstruieren.

Für Hypermedia-Systeme wird häufig die Repräsentation semantischer Netze [Nils1990] verwendet. Eine weitere Möglichkeit der Repräsentation ist die Verwendung von Thesauri als Verallgemeinerung der oben genannten Repräsentationsmöglichkeiten [Trig2001].

2.3.3.2 Lerninhaltsarten

Die Beschreibung der Lerninhalte kann aus verschiedenen Perspektiven betrachtet werden. Dabei sei auf die in Abschnitt 2.3.4 beschriebenen Lerntheorien hingewiesen.

Aus behavioristischer Sicht ist die zentrale Fragestellung, inwieweit Lerninhalte modularisiert und überprüft werden können. Aus kognitivistischer Sicht wird betrachtet, welche Arten von Lerninhalten bestimmte innere Prozesse auslösen und welche Problemlösungsfähigkeiten mit ihnen verknüpft sind. Aus konstruktivistischer Sicht ist die zentrale Fragestellung, wie komplexe Inhalte kontextuell und authentisch in Lernprozessen konstruiert werden. Diese vereinfachte Darstellung zeigt, dass je nach Sichtweise verschiedene Klassifikationen zur Auswahl stehen. In dieser Arbeit wird eine heuristische Klassifikation nach [BaPa1994] gewählt, die verschiedene Perspektiven betrachtet und einen Zusammenhang zwischen Lernzieldimension, Art des Lerninhalts und der Rolle von Akteuren schafft (siehe Abbildung 12).

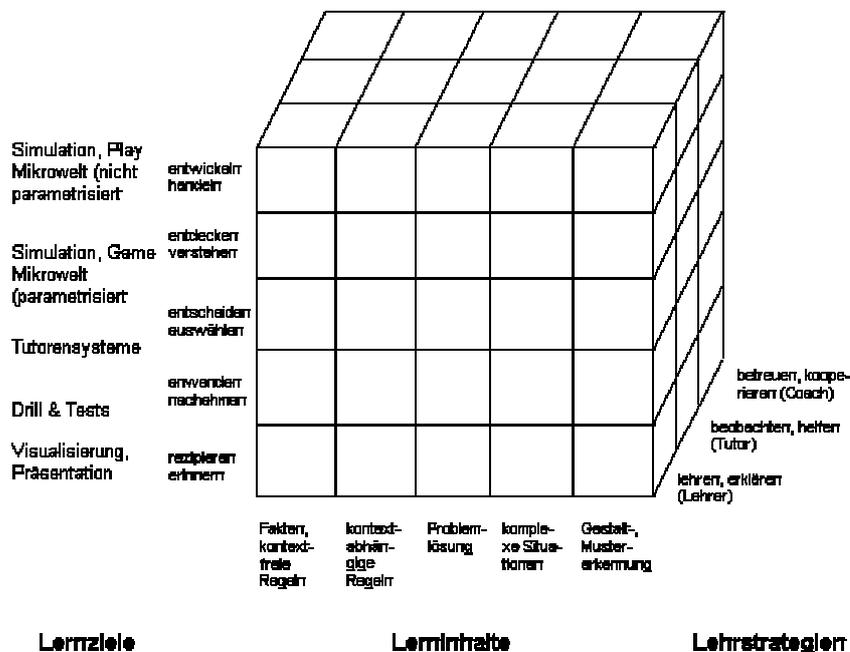


Abbildung 12: Heuristisches Lehr- und Lernmodell nach [BaPa1994]

[BaPa1994, Baum1998] unterscheiden fünf Kategorien von Lerninhalten:

Fakten und kontextfreie Regeln betreffen allgemeingültiges Wissen, das in verschiedenen Kontexten Anwendung finden kann. Der Begriff der Kontextfreiheit ist allerdings missverständlich, da aus konstruktivistischer Sicht Wissen nicht ohne einen Kontext konstruiert werden kann. Allerdings sind auch Fakten als Folge gesellschaftlicher Konstruktionsprozesse zu einem allgemeingültigen Wissen geworden. Unter Verwendung dieser Sichtweise wird die Terminologie daher beibehalten. Mit dem Erlernen von Fakten ist meist eine Lernstrategie verbunden, die sich auf reine Präsentation und Überprüfung beschränkt.

Kontextabhängige Regeln betreffen Regeln, die nur in bestimmten Zusammenhängen sinnvoll anwendbar sind. Das Erlernen der Regeln schließt die Anwendung in einem möglichen Kontext ein, wobei dieser nicht auf den vom Lehrenden vorgesehenen Kontext beschränkt sein muss. Typischerweise sind Übungen und Anwendungsbeispiele mit dieser Kategorie verbunden.

Problemlösung bezieht sich auf die Lösung von Problemen unterschiedlicher Komplexität. Dabei werden Entscheidungsprozesse durchlaufen, die zu einer Problemlösungsstrategie des Lernenden führen.

Bei der *Gestalt- und Mustererkennung* soll von konkreten Situationen abstrahiert werden. Es sollen Muster und Abstraktionen erlernt werden, so dass das erlernte Wissen weiter transferierbar ist.

Bei der Bearbeitung *komplexer Situationen* steht das ganzheitliche Erlernen von Prozessen und deren Zusammenhängen, Ursachen und Wirkungen im Vordergrund. Der Schwerpunkt liegt auf Abstraktion, Wissenstransfer und dem Entdecken neuer Lösungswege.

Diese Klassifikation wird in dieser Arbeit verwendet, um Lerninhalte genauer zu beschreiben.

2.3.4 Lerntheorien

Im Abschnitt zur Begriffsabgrenzung (siehe Abschnitt 2.1) wurde bereits auf verschiedene Sichtweisen bezüglich des Lernbegriffes hingewiesen. Das Design und die Entwicklung von Lernumgebungen hängt maßgeblich von den zugrunde liegenden Lerntheorien ab, die Kenntnisse und Annahmen über den Lernprozess zusammenfassen [Blum1998]. In diesem Abschnitt wird erläutert, welchen Einfluss die Lerntheorien auf die korrespondierenden Lernprozesse haben.

2.3.4.1 Behaviorismus

Forschungsobjekt des *Behaviorismus* ist die Betrachtung objektiv messbaren Verhaltens und geht auf [Wats1930] zurück. Der Behaviorismus basiert auf der Annahme, dass das Verhalten einer Person nicht auf inneren Verarbeitungsprozessen von Wahrnehmungen und Erlebnissen basiert, sondern durch Konsequenzen eines bestimmten Verhaltens [vgl. Eule1992, Kerr2001]. Die Theorie beruht insbesondere auf den Versuchen zur operanten

Konditionierung von Skinner [Skin1954, Skin1958]; eine Übersicht findet sich bei [Eule1992]. Folgt aus einem neu erlernten Verhalten eine positive Konsequenz, so wird dieses Verhalten beibehalten, bei negativen Konsequenzen wird das Verhalten zumindest kurzfristig reduziert. Erfolgt keine Reaktion durch die Umwelt, so wird das Verhalten langfristig gelöscht [Kerr2001]. Aus diesen grundlegenden Annahmen wird gefolgert, dass die Methoden der Belohnung bei positiven Verhaltensänderungen und der Bestrafung bei negativen Verhaltensänderungen zum gewünschten Lernerfolg führen. Der Prozess des Lernens wird als Verhaltensformung (Shaping) angesehen, so dass ein Gesamtziel durch die Verknüpfung einzelner Teilziele erreicht wird. [Kerr2001] weist auf drei wesentliche Mechanismen für Lernumgebungen hin, die aus der Theorie des Behaviorismus abgeleitet werden:

1. Verhaltensänderung und Konsequenz sollten in unmittelbarem Zusammenhang zueinander stehen. Gerade bei neu erlerntem Verhalten sollte eine unmittelbare Rückmeldung erfolgen, damit der Zusammenhang für Lernende deutlich wird und die Verhaltensänderung erhalten bleibt.
2. Verhaltensänderungen sollten durch zwei Strategien erreicht werden. Bei einer positiven Verhaltensänderung sollten Belohnungen erfolgen, bei negativen Verhaltensänderungen sollte möglichst keine Reaktion und somit eine langfristige Löschung erfolgen. Um eine kurzfristige Verhaltensänderung hervorzurufen, werden meist Wiederholungen (als abgeschwächte Form der Bestrafung) eingesetzt. Zu häufige Bestrafungen würden zu negativen Motivationseffekten führen, die den Lernprozess negativ beeinflussen.
3. Die Bekräftigungsrate ist dem Leistungsniveau anzupassen. Das bedeutet, dass nicht bei jeder positiven Verhaltensänderung eine Belohnung erfolgt, sondern nur zu Beginn der Verhaltensänderung. Im Laufe des Lernprozesses muss das Leistungsniveau verbessert werden, um weitere Belohnungen zu erhalten.

Auf diesen grundlegenden Annahmen basiert eine Vielzahl von Lernumgebungen, die unter dem Begriff der *Programmierten Instruktion* zusammengefasst werden [Kerr2001]. Bei dieser Klasse von Lernumgebungen werden die Lerninhalte in elementare Einheiten unterteilt, die dem Lernenden sukzessive präsentiert werden. Im Anschluss an jede Lerneinheit folgt eine Überprüfung des Lernerfolgs durch Fragen, die mit einer hohen Wahrscheinlichkeit von den Lernenden beantwortet werden können. Ist die Lerneinheit erfolgreich abgeschlossen, kann die nächste Lerneinheit bearbeitet werden (Belohnung); bei einem Misserfolg wird die Lernerfolgsüberprüfung wiederholt (Löschung der negativen Verhaltensänderung). Diese Methode wird auch als Drill & Practice bezeichnet (siehe auch Abschnitt 2.2.2).

Der Lehrende steht bei behavioristischen Lernmethoden im Vordergrund. Ihm obliegt die Festlegung von Lernzielen, die damit verbundene Aufteilung der Lerninhalte sowie die Auswahl und Entwicklung der Lernerfolgsüberprüfung.

Die Annahme, dass der Lehrende den Verlauf des Lernprozesses a priori vorhersagen und bestimmen kann, ist als kritisch anzusehen. Ebenso ist die Annahme nicht haltbar, dass Lernprozesse objektivierbar sind und somit bei allen Lernenden ähnlich verlaufen [vgl. Schu1996]. Da behavioristische Systeme keine mentalen Verarbeitungsprozesse einbeziehen, werden individuelle Faktoren vernachlässigt. Gerade bei Lernprozessen, in denen kein beobachtbares Verhalten auftritt, sind behavioristische Systeme nicht einsetzbar [Hase1995]. Ein weiteres Problem ist die starke Atomisierung der Lerninhalte. [Blum1998] folgert, dass kein Zusammenhang zwischen den einzelnen Atomen hergestellt werden

kann, wenn komplexe Problemstellungen vorliegen, die die Anwendung verschiedener Konzepte, Transferleistungen oder das Agieren in einem sozialen Kontext erfordern. Damit sind behavioristische Systeme nur für einfache Lernziele geeignet, wie zum Beispiel die Wiedergabe von Informationen und Fakten oder das Erlernen körperlicher Fertigkeiten [Baum1998].

Dennoch werden behavioristische Systeme weiterhin in vielen Bereichen verwendet. So sind Lernende gerade im Hochschulbereich mit der Methode des Drill & Practice vertraut; diese Methoden werden daher beim Erlernen von einfachen Inhalten und Fakten weiterhin Einsatz finden.

2.3.4.2 Kognitivismus

Der Kognitivismus setzt den Schwerpunkt auf die Erforschung innerer Prozesse des menschlichen Gehirns. Es wird versucht, diese Prozesse zu untersuchen, zu unterscheiden, miteinander in Beziehung zu setzen und in einem theoretischen Modell abzubilden [Baum1998]. Vorrangige Aufgabenstellung ist die Repräsentation menschlicher Denkprozesse wie Erinnerung, Vergessen oder Lernen. Die Art des Lernens unterscheidet sich wesentlich von behavioristischen Sichtweisen. Ziel der Lernprozesse ist nicht das Finden richtiger Antworten, sondern das Erlernen von Methoden und Verfahren zur Lösung von Problemstellungen. Die Rolle des Lernenden unterscheidet sich dadurch maßgeblich: Lernende werden als Individuum verstanden, die äußere Reize aktiv und selbstständig verarbeiten und nicht durch äußere Reize steuerbar sind [vgl. THHL1996].

Lernprozesse werden als Austauschprozesse von Individuen mit der Umwelt verstanden. Basierend auf den zunächst entwicklungspsychologischen Arbeiten von Piaget [Piag1954, Piag1985], steht das Äquilibrationsprinzip, bestehend aus Assimilation und Akkomodation, im Mittelpunkt der Betrachtung: Das Hinzufügen externer Objekte und Zustände in innere Strukturen eines Individuums bezeichnet das Konzept der Assimilation. Akkomodation bezeichnet hingegen den Prozess der Anpassung innerer Strukturen an neue Gegebenheiten der Umwelt unter Veränderung bestehender kognitiver Strukturen [vgl. HoKn1996, Schu1996, Blum1998, GaBe1996]. Handlungsweisen werden zu Schemata zusammengefasst, die Umweltzustände mit Handlungen verknüpfen, die es erlauben, diese Umweltzustände zu modifizieren [HoKn1996]. Nach [Blum1998] ist der Lernprozess die Internalisierung, d.h. die Aufnahme und Verarbeitung von Wissen durch den Aufbau mentaler Modelle und Schemata. Zentrale Aussage des Kognitivismus ist damit, dass neue Informationen immer auf Basis des bereits vorhandenen Wissens interpretiert werden [Kerr2001].

Kognitive Ansätze des didaktischen Designs gehen aufgrund dieser Aussage davon aus, dass die angesprochenen Verarbeitungsprozesse in unterschiedlichen Subsystemen des Gehirns verarbeitet werden. Daraus leiten sich für Lehr- und Lerninhalte jeweils unterschiedliche Verarbeitungsprozesse und damit unterschiedliche Lernmethoden ab. Daher werden verschiedene Ansätze zur Klassifikation von Lerninhalten im nächsten Abschnitt diskutiert.

Eng verbunden mit der kognitivistischen Lerntheorie ist das Prinzip des entdeckenden (explorativen) Lernens [Brun1961]. Dieser Ansatz sieht vor, dass Lernende in einer begrenzten Umgebung selbstständig Informationen auffinden, identifizieren, priorisieren und anwenden. Der Lehrende übernimmt dabei eine Unterstützungsfunktion und hilft den Ler-

nenden bei einzelnen Problemlösungsschritten. Durch diese Strategie soll Neugier, Interesse und damit eine höhere Motivation bei den Lernenden erzeugt werden.

Als Weiterentwicklung zum Behaviorismus betrachtet der Kognitivismus somit insbesondere mentale Prozesse, die zu einem detaillierteren Verständnis von Lernprozessen führen. [Blum1998] und [Baum1998] stellen jedoch als Kritikpunkt fest, dass die Objektivierbarkeit von Problemstellungen nicht immer gegeben ist. Der Kognitivismus geht davon aus, dass jedes Problem objektiv gegeben und repräsentierbar ist und daraus folgend Lösungsschritte gelernt werden können. Dabei wird jedoch übersehen, dass Problemstellungen nicht immer objektivierbar sind und zunächst erkannt und konstruiert werden müssen. Der Prozess der Problemgenerierung wird somit vernachlässigt [Baum1998]. Ein weiterer Kritikpunkt ist die Beschränkung auf geistige Verarbeitungsprozesse. [Baum1998] führt an, dass gerade der Erwerb psychomotorischer Fähigkeiten und Fertigkeiten nach kognitivistischen Prinzipien nur schwer erklärbar ist.

2.3.4.3 Konstruktivismus

Nach Auffassung des Konstruktivismus ist Wissen kein Abbild einer externen Realität, sondern eine Funktion des Erkenntnisprozesses. Wissen wird im Akt des Erkennens von einem erkennenden Subjekt (zum Beispiel einem Lernenden) in einem aktiven Prozess interpretiert [Schu1996]. Wissen wird somit dynamisch während des Konstruktionsprozesses generiert und nicht, wie im Sinne des Kognitivismus, fest gespeichert. [Kerr2001] hebt die Bedeutung der Interaktion hervor: „Die Struktur der Realität, Merkmale wie Regelmäßigkeit, Stabilität oder Kontinuität, existieren nicht an sich, sondern sind Resultat individueller Konstruktionsleistungen in symbolischen Interaktionen“. Der Konstruktivismus verneint nicht die Existenz einer Realität [Baum1998]: nur die Wahrnehmung der Realität ist nicht objektivierbar, d. h., die Realität wird durch eigene Regeln interpretiert [MaVa1987].

Die Bedeutung des Konstruktivismus für Lernprozesse lässt sich durch die Betrachtung des Konzeptes der *Situated Cognition* [Clan1993, Gree1989, Gree1992] genauer untersuchen. [Gree1989, zitiert nach GeMa1994] fassen drei Grundannahmen zusammen.

1. *Situierte Kognition*: Nach diesem Konzept ist Denken immer in physikalischen und sozialen Kontexten situiert. Denken, Wissen und Lernen ist eher die Art, wie ein Handelnder in eine Situation eingebettet ist, als die pure Aktivität eines individuellen Bewusstseins.
2. *Persönliche und soziale Epistemologien*: Denken und Lernen sind weiterhin in Kontexten von Überzeugungen und Bedeutungen von Kognition situiert, so dass Denken und Lernen durch diese Kontexte bestimmt werden.
3. *Konzeptuelle Kompetenz*: Individuen sind zu komplexen und subtilen Prozessen der Wissens- und Bedeutungskonstruktion und Denkfähigkeit imstande. Denken, Lernen und kognitives Wachstum sind die Aktivitäten, mit denen Wissen und Verstehen elaboriert und reorganisiert wird.

[GeMa1994] fassen die Theorie in Form gemeinsamer Positionen zusammen. Wissen ist immer situiert und wird durch das wahrnehmende Subjekt konstruiert. Von wesentlicher Bedeutung ist das in einer Gesellschaft geteilte Wissen (*socially shared cognition*). Schließlich wird situiertes Wissen unter dem Gesichtspunkt der Authentizität analysiert.

[Kerr2001] kritisiert, dass der Konstruktivismus zwar als lernpsychologischer und erkenntnistheoretischer Begriff etabliert ist, der Begriff einer konstruktivistischen Didaktik jedoch nicht eindeutig abgrenzbar ist und damit eine genaue Charakterisierung mediendidaktischer Implikationen schwer fällt. In Anlehnung an [ReMa1994, JoMM1993] lassen sich folgende Implikationen für Lernprozesse festhalten:

- *Konstruktion*: Lernende konstruieren Wissen durch Interpretation in Abhängigkeit von Vorwissen, mentalen Strukturen und bestehenden Überzeugungen. Neue Informationen werden mit Vorwissen verknüpft, um elaborative Wissensstrukturen aufzubauen. [Brow1985] betont die Förderung der Kreativität von Lernenden und die Erhöhung der Motivation bei derartigen Prozessen.
- *Perspektiven*: Auf Grundlage kollaborativer Prozesse von Lehrenden und Lernenden werden Bedeutungen ausgehandelt. Das bedeutet, dass der Lehrende keine objektive Realität präsentiert, sondern verschiedene Perspektiven und Sichtweisen und damit auch unterschiedliche Lernergebnisse möglich sind.
- *Kontext*: Lerninhalte müssen in Bezug zu einem relevanten Kontext stehen, der sich auf Vorwissen oder Erfahrungen der Lernenden bezieht. Zur Reflexion des Lernhandelns sollten metakognitive Fertigkeiten eingesetzt werden.
- *Lernumwelt*: Der Schwerpunkt liegt nicht mehr in der Planung des Unterrichts, sondern in der Gestaltung von Lernumgebungen oder -umwelten (learning environments), die oben genannte Perspektiven einbeziehen und weitere Unterstützungswerkzeuge (z. B. cognitive tools) bieten [vgl. Brow1995, Schu1996, DuLJ1993].

2.3.4.4 Zusammenfassung und Folgerungen

Das Beschreibungsmerkmal der Lerntheorien ist ein zusammenfassendes Merkmal, das einen Überblick über eine Lernumgebung und Designprinzipien gibt. Die Ausgestaltung der Lernumgebung muss jedoch durch weitere Merkmale beschrieben werden. Die Annahmen und die Auswirkungen auf verschiedene Merkmale bzw. Designentscheidungen sind in Tabelle 1 in Anlehnung an [Blum1998] zusammenfassend dargestellt.

Tabelle 1: Zusammenfassung der Lerntheorien [Blum1998]

| | Behaviorismus | Kognitivismus | Konstruktivismus |
|--|--|---|--|
| Hirn ist | Black Box | Informations- verarbeitendes Gerät | informell geschlossenes System |
| Wissen wird | gelagert | verarbeitet | konstruiert |
| Wissen ist | korrekte Input- Output-Relation | adäquater interner Verarbeitungsprozess | mit einer Situation operieren zu können |
| Lernziele | richtige Antworten | richtige Methoden zur Antwortfindung | komplexe Situationen bewältigen |
| Lern- theoretisches Paradigma | Stimulus-Response | Problemlösung | Konstruktion |
| Strategie | lehren | beobachten, helfen | kooperieren |
| Lehrender ist | Autorität | Tutor | Coach (Spieler) Trainer |
| Feedback | extern vorgegeben | extern modelliert | intern modelliert |
| Interaktion | vorgegeben | dynamisch in Abhängigkeit des externen Lernmodells | selbstreferentiell, zirkulär, strukturdeterminiert |
| Programmmerkmale | festgelegter Ablauf, quantitative Zeit- und Antwortstatistik | dynamischer Ablauf, vorgegebene Problemstellung, Antwortanalyse | dynamisch vernetzte Systeme, keine vorgegebenen Probleme |
| Informationstechnologisches Paradigma | Lernmaschine | Künstliche Intelligenz | Soziotechnische Umgebung |
| Idealer Softwaretypus | Drill & Practice, Tutorielle Systeme | Intelligente Tutorielle Systeme, CBT, Adaptive Systeme | Simulationen, Mikrowelten |

Abschließend lässt sich folgern, dass jede Lerntheorie für bestimmte Lerninhalte geeignet sein kann. Vielmehr werden für verschiedene Inhalte, Aktoren und Kontexte unterschiedliche Methoden benötigt. So können nur Empfehlungen ausgesprochen werden, die dann an den Kontext angepasst werden müssen.

So wird nach [JoMM1993] empfohlen, Lernumgebungen gerade im universitären Kontext und mit den damit verbundenen komplexen Lerninhalten an konstruktivistischen Prinzipien auszurichten. [MaGR1997] stellen folgende Anforderungen an Lernumgebungen:

- *Komplexität der Ausgangsprobleme*: Den Ausgangspunkt für ein interessantes und intrinsisch motiviertes Lernen sollte ein komplexes Ausgangsproblem bilden, das im Idealfall mehrere Lösungsmöglichkeiten zulässt. Es sollte eine Herausforderung für den Lernenden darstellen und aus seinem Erfahrungsbereich stammen. Nach Möglichkeit soll der Lernende ein Gefühl des Besitzes für ein Problem entwickeln. Ein komplexes Ausgangsproblem kann beispielsweise durch einen komplizierten Entscheidungsfall oder eine Gestaltungs- und Beurteilungsaufgabe gegeben sein.
- *Authentizität und Situiertheit*: Die Lernumgebung sollte den Lernenden mit realistischen Problemen, die in authentische Situationen eingebettet sind, konfrontieren. Auf diese Weise wird für das Wissen ein Rahmen und ein Anwendungskontext zur Verfügung gestellt. Die Situiertheit legt den Fokus, dass Lernsituationen sich an für die Lernenden bereits bekannten Sachverhalten orientieren.
- *Multiple Perspektiven und Kontexte*: Damit der Lernende das Wissen auf andere Bedingungen und Zusammenhänge übertragen kann, ist es bedeutsam, dass das Wissen nicht auf einen Kontext beschränkt ist, sondern es auf andere Problemzusammenhänge übertragen werden kann. Durch diesen Perspektivenwechsel lernt er nicht nur eine kritische Auseinandersetzung mit den Inhalten, sondern auch verschiedene Standpunkte zu betrachten und in Beziehung zu setzen und Inhalte unter variierenden Aspekten zu bearbeiten. Dieser Prozess ist besonders bedeutsam, um träges Wissen zu vermeiden und die flexible Wissensanwendung zu fördern.
- *Artikulation und Reflexion*: Problemlösungsprozesse sollten reflektiert und artikuliert werden, um kontextgebundenes Wissen zu vermeiden. Auf diese Weise wird zudem die Abstrahierung von Wissen gefördert. Abstrahiertes Wissen unterscheidet sich vom abstrakten Wissen dadurch, dass es mit Situationen verbunden und somit anwendbar ist. Des Weiteren fördern Artikulation und Reflexion den Prozess der Metakognition, also das Nachdenken über die eigenen Lernprozesse. Der Lernende ist besser in der Lage, Wissen zu strukturieren und sich allgemeine Problemlösungsstrategien anzueignen und zu modifizieren.
- *Lernen im sozialen Kontext*: Ein wesentlicher Bestandteil der Lernsituation sollte in der Möglichkeit bestehen, im sozialen Austausch zu lernen und sich auszutauschen. Gemeinsames und kooperatives Lernen sollte ebenso gefördert werden wie Lerngruppen. Die Entwicklung und Einbindung der Lernenden in eine Expertenkultur wird als wichtiger Bestandteil für erfolgreiches Lernen gesehen.

2.3.5 Aktoren

Die Beschreibung von Aktoren des Lernprozesses kann aus verschiedenen Perspektiven betrachtet werden. Dabei ist die Zielsetzung dieser Beschreibung, den Lernprozess an bestimmte Merkmale der Beteiligten anzupassen.

2.3.5.1 Rolle

Rollen fassen Eigenschaften, Charakteristika und Anforderungen an einen Akteur in Form von Stereotypen zusammen. Eine einfache Unterscheidung ist die Aufteilung der Rollen in Lehrende und Lernende. Diese klassische Unterscheidung betrachtet nur das Merkmal der Verantwortlichkeit. Detailliertere Unterscheidungen bedeuten jedoch auch entsprechende Kompetenzen oder Verhaltensweisen. Die Rolle des Lehrenden wird zum Beispiel häufig in die Rollen Lehrender, Tutor, Coach und Moderator aufgeteilt. Dies impliziert didaktische Methoden, in der der Lehrende bestimmte Verhaltensweisen annimmt. Diese Implikationen ermöglichen jedoch keine genaue Identifizierung oder Klassifikation der Beteiligten. Daher wird an dieser Stelle ein Rollenkonzept gewählt, das sich an den Aufgaben während der Lern- und Entwicklungsprozesse orientiert. Die genauere Beschreibung der Aktoren wird durch zusätzliche Merkmale vorgenommen.

Die Beteiligten am Lernprozess können durch die vorrangigen Aufgaben während des Lernprozesses beschrieben werden. Tabelle 2 fasst die Hauptaufgaben der einzelnen Rollen in Anlehnung an [Paqu2001] zusammen. Ähnliche Klassifikationen finden sich bei [SaSc1997, Kerr2001].

Tabelle 2: Rollen in Lernprozessen

| Rolle eines Aktors | Aufgabe |
|--------------------|---|
| Lernender | Wissenserwerb |
| Lehrender | Unterstützung des Lernprozesses und des Lernenden |
| Entwickler | Entwicklung und Wartung von Lernumgebungen |
| Manager | Management und Administration des Entwicklungs- und Lernprozesses |
| Domänenexperte | Bereitstellung der Inhalte |
| Mediendesigner | Produktion von Medienangeboten |

Insbesondere die Aufgaben und Kompetenzen von Lehrenden und Lernenden müssen zusätzlich charakterisiert werden, um die Aussagekraft und Anwendbarkeit des Rollenkonzeptes zu verbessern. Dazu werden im Folgenden weitere Merkmale spezifiziert.

2.3.5.2 Lernende

Die Entwicklung computerunterstützter Lernumgebungen hat das Ziel, Lernprozesse zu verbessern. Gerade die individuellen Anpassungsmöglichkeiten an Lernende stehen dabei im Vordergrund. Die zentrale Fragestellung ist, welche Informationen über den Lernenden zur Verfügung stehen müssen, um eine individuelle Anpassung vornehmen zu können. Die kontroverse Diskussion dieser Thematik wird im Abschnitt 4.4.4.1 (Entwicklung von Standards für Lernermodelle) aufgegriffen. Im folgenden Abschnitt werden exemplarisch Attribute zur Charakterisierung von Lernenden genannt, der Adaptationsprozess ist nicht Bestandteil dieser Arbeit.

Die Beschreibung lässt sich auf verschiedenen Ebenen durchführen. Es werden Daten über den individuellen Lernenden erfasst, wobei die Daten für bestimmte Lerngruppen (bzw. Zielgruppen) häufig aggregiert werden. In Anlehnung an [Kerr2001] können folgende Klassen zur Charakterisierung der Lernenden (Zielgruppe) unterschieden werden:

Für eine grobe Einschätzung der Zielgruppe können zunächst *soziodemographische Faktoren* herangezogen werden. [Kerr2001] nennt dabei folgende Attribute: Gruppengröße, geographische Verteilung, Alter, Geschlecht, Schulabschluss, PC-Kenntnisse und Kaufbereitschaft für Lernumgebungen.

Das *Vorwissen* beschreibt Lerninhalte, die ein Lernender bereits erlernt hat. Dabei wird unterschieden, wann und in welcher Komplexität Lerninhalte behandelt wurden, da offensichtlich nicht auf beliebige Lerninhalte aus der Lernhistorie zurückgegriffen werden kann oder zumindest eine Wiederholung sinnvoll erscheint. Lerninhalte können, wie in 2.3.3.2 diskutiert, z. B. durch Lernziele oder auch sogenannte Skills charakterisiert werden. Inhalte, die spezifisch für den Umgang mit Lernumgebungen benötigt werden, kommen hinzu. Diese werden von [Kerr2001] als *Einstellungen und Erfahrungen* bezeichnet. Diese Unterscheidung ist allerdings nicht notwendig, da Einstellungen und Erfahrungen als kognitives und affektives Vorwissen bezüglich bestimmter Lernmethoden und Technologien angesehen werden können.

Lernpräferenzen beschreiben Methoden und Konzepte des Lernens, die von Lernenden bevorzugt verwendet werden. Dabei geht [Kerr2001] auf Lerngewohnheiten ein, mit denen Lernende vertraut sind; diese beschreiben die Situationen des Lernens. Die Durchführung eines Studiums und korrespondierende Lernprozesse unterscheiden sich zum Beispiel wesentlich bei einem Abiturienten, der ohne Verzögerung eine Universität besucht und einem Lernenden, der ausgehend von einer Berufspraxis mit geringen Weiterbildungsanteilen ein Studium aufnimmt. Häufiger wird auch der *Lernstil* von Lernenden diskutiert. Der Lernstil bezeichnet Persönlichkeitsmerkmale bezüglich der Formen des Lernens, mit denen größtmögliche Lernerfolge erwartet werden. [JoGr1993] unterscheiden dabei weiterhin nach Cognitive Controls, Cognitive Styles und Learning Styles, die im Folgenden in Anlehnung an [Blum1998] kurz umrissen werden. Ein Schwerpunkt wird auf die Implikationen gelegt, die sich für Lernprozesse ergeben.

Cognitive Controls beschreiben Eigenschaften des Lernenden, die die Wahrnehmung von Umweltreizen betreffen. Maßgebliches Unterscheidungskriterium ist der Einfluss der Umwelt auf die Wahrnehmung und das Verständnis von Informationen. Dabei unterscheidet man feldabhängige und feldunabhängige Lernende. Feldabhängige Lernende neigen zu einer wenig reflektierten Akzeptanz präsentierter Sachverhalte, benötigen ein hohes Maß

an Führung und bevorzugen Situationen mit hoher sozialer Interaktion, während feldunabhängige Lernende Informationen eher reorganisieren und restrukturieren sowie konzeptueller arbeiten, dafür jedoch weniger zu sozialer Interaktion tendieren.

Cognitive Styles beschreiben die Vorgehensweise eines Lernenden bei der Organisation und Strukturierung von Informationen. Eine serielle Vorgehensweise bedeutet, dass sich Lernende zunächst eher auf Details konzentrieren und dann zu einem Gesamtzusammenhang aggregieren (Bottom-Up), während bei einer holistischen Vorgehensweise eher vom Gesamtzusammenhang ausgegangen und dieser später in Details vertieft wird.

Der *Learning Style* im engeren Sinne betrifft bevorzugte Lernformen, die insbesondere in Intelligenten Tutoriellen Systemen Berücksichtigung finden. Dabei wird in der Literatur eine Vielzahl an Klassifikationen diskutiert, die jeweils unterschiedliche Merkmale zur Kategorisierung heranziehen und daraus verschiedene Stereotypen ableiten, exemplarisch siehe [Kolb1981, Blum1998, HaSt1993, Schu1996]. So kategorisiert [Kolb1981] nach den Kriterien Verständnis (abstrakt, konkret) und Transformation (aktiv, reflektiv). Daraus ergeben sich vier Klassen von Lernenden:

Tabelle 3: Lerntypen nach [Kolb1981]

| Transformation | Aktiv | Reflektiv |
|-----------------------|--------------|------------------|
| Verständnis | | |
| Abstrakt | Converger | Assimilator |
| Konkret | Accomodator | Diverger |

Lernstile können zudem bezüglich der Modalität klassifiziert werden [Paiv1971, LDKG1974, FrBa1997]. Dabei unterscheidet man zwischen Visualisierer und Verbalisierer. Verbalisierer bevorzugen die Aufnahme von Informationen über eine textuelle Codierung, während Visualisierer eine bildliche Codierung bevorzugen. Diese Klassifikation ist eine wesentliche Simplifizierung, die sich nur auf kognitive Lernziele beschränkt. Andere Codierungen, weitere Medienformen sowie deren Kombination und Wechselwirkung werden nicht bedacht.

Es zeigt sich, dass sich der Begriff Lernstil auch auf jedes andere in diesem Kapitel angesprochene Merkmal (Lerninhalt, Lernziel, Lernort etc.) beziehen kann und keine einheitliche Klassifikation besteht. Daher ist die umfassende Charakterisierung der Lernenden sinnvoll, während darauf aufbauende Stereotypenbildung und die damit verbundene Anpassung abhängig vom jeweiligen Kontext entschieden werden muss.

Die *Lernmotivation* beschreibt den Grund, warum ein Lernender an einem bestimmten Inhalt interessiert ist und ist somit eng mit inhaltlichen Lernpräferenzen verbunden. Intrinsisch motivierte Lernende lernen aus Interesse am Lerngegenstand, während extrinsisch motivierte Lernende durch äußere Anreize motiviert werden (z. B. Weiterbildungsmaßnahme mit Aufstieg und entsprechender Einkommensverbesserung, Belohnungen im behavioristischen Sinne) [Kerr2001]. Der *ARCS-Ansatz* (Attention, Relevance, Confidence, Satisfaction) betrachtet vier maßgebliche Faktoren zur Erklärung der Motivation [Kell1987, Blum1998] und folgert daraus, welche Prozessschritte im Instruktionsdesign zu beachten sind. Generell lässt sich feststellen, dass die Motivation von Lernenden nur

schwer zu formalisieren ist. Ein großes Problem im Kontext der Lerntechnologien ist dabei der Effekt der Motivation durch neue und damit interessante Technologien, der getrennt von einer inhaltsorientierten Motivation gesehen werden muss. Der *Hawthorne-Effekt* [Schu1996] beschreibt die Motivation von Lernenden, die durch die Verwendung neuer Technologien verursacht ist und damit die Interpretation von Evaluationen erschwert.

2.3.5.3 Lehrende

Gerade im Forschungsbereich Intelligenter Tutorieller Systeme wird ein Schwerpunkt auf die Lernermodellierung und Anpassung von Lernumgebungen an den Lernenden gelegt. Dabei wird häufig die Analyse und Auswahl der Lehrenden vernachlässigt. Häufig wird die Spezifikation von Lehrenden auf persönliche Daten und die Einordnung in ein Rollenkonzept beschränkt. Ein solches Rollenkonzept nach [Paqu2001] unterscheidet die Rollen:

Tabelle 4: Rollen der Lehrenden

| Rolle | Aktivitäten |
|---|---|
| Trainer (Coach) | Training und Leitung des Lernprozesses |
| Berater (Advisor) | Methodische Beratung im Lernprozess |
| Prüfer (Evaluator) | Überprüfung des Lernerfolgs |
| Techniker (Helper in using the environment) | Technische Unterstützung zur Nutzung der Lernumgebung |
| Analytiker (Diagnostic Producer) | Analyse der Lernprozesse und der Lernleistung |
| Teamleiter (Team Animator) | Motivation und Gruppenbildungsprozesse |

Die Rolle impliziert jeweils einen bestimmten Aufgabenbereich des Lehrenden. Dieses beschränkte Lehrermodell lässt jedoch keine Abstimmung von Lehr- gegenüber Lernerfahrung oder Lehrpräferenzen gegenüber Lernpräferenzen zu. Daher werden die in Abschnitt 2.3.5.2 beschriebenen Charakterisierungen auch für Lehrende spezifiziert und genutzt.

2.3.6 Interaktion

Die Interaktion beschreibt mögliche Kommunikationsformen zwischen Lehrenden, Lernenden und Lernumgebung. Dabei umfasst die Interaktionskomponente nicht nur Kommunikationssysteme, sondern ebenfalls Systeme zur Kooperation und Koordination.

Zunächst wird betrachtet, welche *Aktoren* an Kommunikationsprozessen in einer Lernsituation beteiligt sind. Dabei muss im Allgemeinen das Verhältnis zwischen Lehrenden, Lernenden und Lernumgebung untersucht werden. Die Lernumgebung kann jedoch auch als Hilfsmittel für den Lehrenden und Lernenden gesehen werden, um Kommunikationsprozesse zu unterstützen. So kann zum Beispiel eine automatisierte Fragensauswertung als Kommunikationsfunktion oder auch als Hilfsmittel des Lehrenden interpretiert werden.

Daher wird die Lernumgebung bzw. das Kommunikationssystem nur als Vermittler, nicht aber als eigenständiger Akteur angesehen.

Das wichtigste Unterscheidungskriterium ist die *Richtung* der Kommunikation: Unidirektionale Kommunikation verläuft einseitig. Diese Beschränkung kann einerseits methodenimmanent sein (z. B. Vorträge), andererseits kann sie durch technische Beschränkungen bedingt sein (z. B. Fernsehen, Video). Bidirektionale Kommunikation verläuft dagegen in zwei Richtungen, d.h., dass eine wechselseitige Kommunikation möglich ist. Gerade in Lernprozessen wird eine bidirektionale Kommunikation in den meisten Methoden als unerlässlich angesehen. Die Möglichkeit der bidirektionalen Kommunikation wird als besonderer Vorteil neuerer Entwicklungen des CUL (z. B. internetunterstützte Lernumgebungen) angesehen. So betont [Schu2001], dass gerade die Bildung von Wissensgemeinschaften aufgrund der Möglichkeit des Gruppenlernens, der Reflexion und der sozialen Einbindung von besonderer Bedeutung ist. Die Bedeutung der bidirektionalen Kommunikation spiegelt sich ebenfalls in neuen Technologien wider. So werden Medien, die ursprünglich nur für unidirektionale Interaktion konzipiert waren, um einen Rückkopplungskanal erweitert. Als Beispiel sei *Interactive Television* [Bate2001] genannt. Fernsehen wird in diesem Fall durch den Benutzer gesteuert und beeinflusst. Lerninhalte, die in dieser Form präsentiert werden, können somit beeinflusst werden, um auf entsprechende Lernpräferenzen abgestimmt zu werden.

Ein weiteres Kriterium, das Lernprozesse maßgeblich beeinflusst, ist der *Lernzeitpunkt*. Synchroner Kommunikation bedeutet, dass Lehrende und Lernende gleichzeitig anwesend sind, während asynchrone Kommunikation zu unterschiedlichen Zeitpunkten durchgeführt wird. Ebenso kann für Kommunikationsprozesse eine räumliche Unterscheidung vorgenommen werden (siehe Abschnitt 2.3.2.2). Des Weiteren kann die Kommunikation geplant und damit vorhersehbar sein oder spontan ohne vorherige Planung erfolgen. Abbildung 13 zeigt eine Übersicht über Beispiele dieser Kommunikationsformen und zeigt mögliche Kommunikationsausprägungen. Dabei sind die einzelnen Ausprägungen nicht notwendigerweise computerunterstützt.

| Ort \ Zeit | Zeit | | |
|--------------------------------|---------------------|--------------------------|--------------------------------------|
| | Gleich | Verschieden vorhersehbar | Verschieden nicht vorhersehbar |
| Gleich | Gemeinsame Sitzung | Schichtarbeit | Schwarzes Brett |
| Verschieden vorhersehbar | Video-konferenz | E-Mail | Gemeinsames Verfassen von Dokumenten |
| Verschieden nicht vorhersehbar | Mobilfunk-konferenz | Bulletin Board | Vorgangsbearbeitung |

Abbildung 13: Beispiele für Kommunikationsprozesse

Gemeinsame Sitzungen entsprechen klassischen Präsenzveranstaltungen, in denen Lehrende und Lernende gleichzeitig anwesend sind. Die Interaktion ist in diesem Fall meist bidirektional. Durch die Möglichkeit, direkt auf Reaktionen der Akteure eingehen zu können,

wird diese Interaktionsform weiterhin Anwendung finden. Die Möglichkeit der Computerunterstützung ist durch Entscheidungsunterstützende Systeme [Lust2001] oder Präsentationssysteme gegeben.

Schichtarbeit und Interaktionen über ein *Schwarzes Brett* implizieren, dass zwar der Ort gleich, der Zeitpunkt der Interaktion jedoch verschieden ist. Die Interaktion ist in der Regel unidirektional. Diese Art der Interaktion ist in Lernprozessen nur selten ausgeprägt. So können als *Schichtarbeit* Gruppenlernprozesse angesehen werden, in die Aktoren einer Lerngruppe Teilaufgaben sukzessive bearbeiten und nur der Arbeitsfortschritt kommuniziert wird.

Videokonferenzen und die derzeit noch weniger verbreiteten *Mobilfunkkonferenzen* sind synchrone Interaktionsausprägungen. Dabei wird versucht, Präsenzlernsituationen durch den Technologieeinsatz zu substituieren. Die Wirkung der Interaktion unterscheidet sich jedoch wesentlich von Präsenzinteraktionen.

Bulletin Boards und *E-Mail-Anwendungen* werden bei asynchronen, verteilten Lernprozessen eingesetzt und können sowohl uni- also auch bidirektional verwendet werden.

Mehrbenutzereditoren und *Instant Messaging-Systeme* sind schließlich für verteilte, synchrone Lernprozesse geeignet.

Ein komplexeres Unterscheidungskriterium ist der *Grad der Anpassung*. Passive Interaktion bedeutet, dass Kommunikationsprozesse vordefiniert sind und einem starren Ablauf folgen. Adaptive Interaktion bedeutet, dass der Interaktionsprozess an Charakteristika der Lernenden angepasst wird.

Bisher wurden Unterscheidungsmerkmale betrachtet, die für Kommunikation im engeren Sinne gelten. Erweitert man den Kommunikationsbegriff und bezieht dessen Ziele ein, so kann eine Unterscheidung aufgrund der *Kommunikationsintensität* getroffen werden [vgl. BoSc1995, BiKö2001]. Es ist zu beachten, dass diese Betrachtung des Kommunikationsbegriffes eine Überschneidung zur didaktischen Methodik einschließt, da zum Beispiel im Rahmen des Computer Supported Cooperative Learning bestimmte Kommunikationsprozesse Hauptbestandteil des didaktischen Konzepts sind.

- *Information*: Absender und Empfänger kommunizieren meist unidirektional. Ziel dieser Interaktion ist der Transfer von Informationen zwischen mehreren Aktoren.
- *Koordination*: Eine Abstimmung eines Arbeits- oder Lernprozesses wird zeitlich oder sachlogisch koordiniert, die Kommunikation erfolgt bidirektional.
- *Kooperation*: Die Zusammenarbeit erfolgt durch periodische, bidirektionale Kommunikation, Teilergebnisse werden abgestimmt und abschließend zusammengefasst [vgl. Pani2001, Craw2001].
- *Kollaboration*: Die Zusammenarbeit erfolgt durch regelmäßige bidirektionale Kommunikation, das Ziel und Ergebnis der Gruppenarbeit wird gemeinsam festgelegt und verantwortet.

Es ist zu beachten, dass Interaktionsformen immer Bestandteil einer didaktischen Methode sind. In jeder Methodenspezifikation müssen Interaktionsmöglichkeiten und deren Einbindung in den Lernprozess beschrieben werden. Dabei hängt die Auswahl der Interaktionen

von den bereits beschriebenen Faktoren wie Kontext, Lernpräferenzen oder Lernmotivation ab. Die Zielsetzung bzw. die Intensität der Kommunikation muss mit der Zielsetzung der didaktischen Methode übereinstimmen. Die Auswahl einer Interaktionsapplikation muss daher immer in Verbindung mit der Methodenauswahl betrachtet werden. Umfasst zum Beispiel eine didaktische Methode eine Diskussionsphase, so können daraufhin mögliche Kommunikationstechnologien ausgewählt werden (Bulletin Board, Videokonferenz).

2.3.7 Präsentation

Die *Präsentationsform* beschreibt, in welcher Form Lerninhalte aufbereitet werden. Dabei unterscheiden [BoSc1995] zwischen kontinuierlichen und diskreten Medien sowie realen gegenüber synthetisierten Medienformen (Abbildung 14).

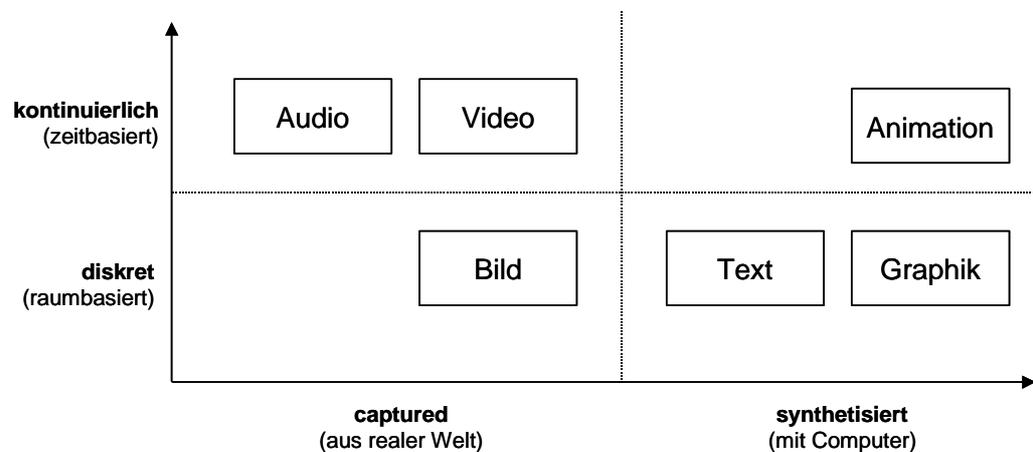


Abbildung 14: Präsentationsformen nach [BoSc1995]

Eine weitere mögliche Unterscheidung bezieht den *Grad der Interaktion* in die Präsentationsform ein. Passive Präsentationen beschränken sich auf eine vordefinierte Abfolge zur Informationsdarstellung, während interaktive Präsentationen die Anpassung an den Lernenden zulassen. So kann Geschwindigkeit, Medienform oder Inhalt variiert werden. Des Weiteren können Annotationen oder die Anpassung bezüglich der Lernerpräferenzen vorgenommen werden.

Die Auswahl des Mediensystems ist wie auch die Wahl der Interaktionsform in Verbindung mit der Wahl einer didaktischen Methode zu sehen. [Kerr2001] beschreibt verschiedene Ansätze der Medienselektion. So verfolgt der Ansatz von [Merr1994] eine algorithmische Auswahl des Präsentationsmediums in Abhängigkeit von der Art der Lernziele. [Kerr2001] merkt jedoch an, dass dieses Selektionsverfahren auf einzelne Medien, nicht jedoch auf multimediale Angebote anwendbar ist. Des Weiteren wird der Kostenrahmen nicht beachtet. Weitere algorithmische Ansätze zur Medienselektion werden im Abschnitt 3.4.1 zum Instruktionsdesign diskutiert. Den Ansätzen gemein ist die Aussage, dass erst abhängig von den Lernenden und von der Art der Lerninhalte eine optimale Präsentationsform bestimmt werden kann.

Ein weiterer Ansatz nach [Clar1994] diskutiert die Wirksamkeit von Medien. Die Schlussfolgerung dieser Untersuchungen ist, dass eine Überlegenheit bestimmter Medienformen nicht nachgewiesen werden kann. Der Lernerfolg lässt sich also prinzipiell mit jeder Medienform erreichen. [Kerr2001] folgert, dass nicht die Auswahl, sondern vielmehr die Kombination in den Vordergrund gestellt werden sollte. Die Verzahnung und Integration von Medien in einer Lernumgebung wird hier als kritischer Erfolgsfaktor angesehen. Dabei werden insbesondere hybride Lernarrangements gestaltet, die kein einzelnes Medium auswählen, sondern eine didaktische und mediale Vielfalt bieten. Die Auswahl des Mediums kann dann den Lernenden überlassen werden.

Eine sinnvolle Auswahl und Kombination von Medien kann jedoch nicht a priori vorgenommen werden. Daher muss in Entwicklungsprozessen von Lernumgebungen eine verstärkte Benutzerpartizipation erfolgen. Die Nutzung und Wirksamkeit der Präsentationsformen sollte bereits während der Entwicklung evaluiert werden. Durch die Einbeziehung der Lernenden kann dann ein ausgewogenes Angebot und eine entsprechende Medienvielfalt geboten werden, die im späteren Lernprozess auch angewendet wird.

2.3.8 Didaktische Methoden

Der Begriff einer didaktischen Methode ist nicht einheitlich definiert. So wurde bereits erläutert, dass auf unterschiedlichen didaktischen Ebenen offensichtlich unterschiedliche Methoden verwendet werden. In diesem Abschnitt werden didaktische Methoden im Sinne der Planung von Kurs- und Lerneinheiten erläutert (D- und E-Ebene, siehe Abschnitt 2.3.1.1).

Auf diesen Ebenen lassen sich unterschiedliche Vorgehensweisen betrachten: Zum einen die Darstellung und Begründung globaler didaktischer Strategien, zum anderen die Beschreibung und Begründung prototypischer Klassen didaktischer Handlungen. In diesem Abschnitt wird die zweite Alternative gewählt; die umfassendere Beschreibung globaler didaktischer Strategien wird im Abschnitt 3.4 aufgegriffen.

Ein weiterer Bereich ist die Modellierung didaktischer Methoden. Gerade im Bereich des Instruktionsdesigns (siehe 3.4.1) und dem Bereich der Intelligenten Tutoriellen Systeme (siehe 2.2.2.4) werden verschiedene Modelle und Repräsentationen verwendet [vgl. Murr1999]. Ein umfangreicheres *Instructional Model* verwendet [PaAC1999, Paqu2001]. In diesem Modell werden Instruktionsprinzipien, Lernereignisse, Instruktionsszenarios, Lernaktivitäten, Lerninstrumente und Aktoreneigenschaften erfasst. Hinzu kommt eine Beschreibung der Lerninhalte, Lernmaterialien und der Übermittlung. Diese Modell umfasst also eine ausführlichere Beschreibung und lässt die Spezifikation unterschiedlicher Szenarios zu. Die Vergleichbarkeit der Modelle, die zudem meist in einem spezifischen Kontext verwendet werden, ist jedoch nicht gegeben.

Da sich in diesem Bereich keine einheitlichen Klassifikationen existieren, werden didaktische Methoden anhand vergleichbarer Kriterien beschrieben. In [Schm2000, FeSc2001] werden die Kriterien Lernbedarf, Domänengliederung, Zeitpunkt, Aufgabenträger, Kommunikation und Kooperation zur Charakterisierung von Lehr- und Lernsituationen herangezogen und für typische Formen der Hochschullehre (Vorlesung, Übung, Seminar und Selbstlernen mit Lehrbuch) diskutiert. Als weiteren Ansatz sei die Beschreibungssystema-

tik von [FIBS1978, Flec1996, Prei1998] genannt, die Ausgangspunkt für die nachfolgenden Ausführungen ist. Methoden werden durch didaktische Prinzipien, durch beteiligte Akteure (Elemente) und durch den Ablauf in Form eines Phasenmodells beschrieben.

Im Zusammenhang computerunterstützten Lernens muss diese Beschreibungssystematik noch erweitert werden. Informationsobjekte beschreiben elektronische Ressourcen, die in dieser Methode verwendet werden können; die Präsentationsform der Ressourcen ist im Allgemeinen allerdings nicht vorzuschreiben. Kommunikationsobjekte beschreiben computerunterstützte Anwendungen, die die Interaktion der Akteure untereinander und mit dem System unterstützen.

Im Folgenden werden nun exemplarisch Methoden dargestellt, die im universitären Kontext angewendet werden und die eine maßgebliche Entwicklungsrichtung im Forschungsbereich der Lerntechnologien repräsentieren. Das Essener-Lern-Modell unterstützt derzeit verschiedene Methoden, die in Abschnitt 5.5 weitergehend erläutert werden.

2.3.8.1 Fallmethode (Fallstudie)

Bei der *Fallmethode* bearbeiten Lernende einzeln oder in Gruppen rekonstruierte Praxisfälle, um sich Wissen über die betreffende Praxis anzueignen und Urteils- und Entscheidungsfähigkeit auszubilden. Die Lernenden versetzen sich in die Rolle von real handelnden Personen, die komplexe Zusammenhänge überblicken müssen. Das verwendete Fallmaterial sollte den Tatbestand möglichst genau und überschaubar dokumentieren. Fallstudien vermitteln Handlungs- und Entscheidungswissen.

Die didaktischen Prinzipien dieser Methode sind praxisnahes, problemlösendes Lernen, das Lernen an komplexen Sachverhalten und Vergangenheitsbeispielen, die insbesondere offene Entscheidungsalternativen besitzen.

Tabelle 5: Phasen der Fallmethode

| Phase | Beschreibung |
|--------------------|--|
| Vorbereitungsphase | Einführung der Methode und des Themenbereichs |
| Rezeptionsphase | Beschaffung und Bearbeitung der Informationen |
| Interaktionsphase | Vergleich von Problemdefinitionen, Prüfung der Lösungsalternativen, Entscheidungsfindung |
| Anwendungsphase | Vergleich der Entscheidungen mit realen Situationen |
| Bewertungsphase | Diskussion der Lösungsalternativen, Entscheidungsfindung im Plenum |

Es ist offensichtlich, dass an verschiedenen Stellen Anknüpfungspunkte für einen sinnvollen Technologieeinsatz zu finden sind. So können Szenarien im Sinne der *Anchored Instruction* [CTGV1990, CTGV1993] multimedial präsentiert werden. Eine aufwendigere, aber in hohem Maße praxisfördernde Maßnahme ist die Integration von Arbeits- und Lernprozessen [AdKP1998a, AdKP1998b, AdKP1999]. Dabei werden zum Beispiel in der betrieblichen Weiterbildung Arbeitsprozesse und damit verbundene Problemstellungen durch

die Arbeit in Übungssystemen simuliert. Somit arbeiten die Lernenden an einem realitätsnahen System und können nach der Weiterbildung im gewohnten System produktiv arbeiten. Diese Integration wird auch von [ReRa2001] durch die Verwendung von EPSS unterstützt.

2.3.8.2 Fernunterricht (*Fernkurs, Fernstudium, Korrespondenz-Unterricht*)

Fernunterricht ist historisch eine der ersten Implementierungen des computerunterstützten Lernens. Dabei wurde die Methodik des traditionellen Fernstudiums durch elektronische Kommunikationsformen ersetzt. Bei dieser Methode eignen sich Lerner durch Lektüre von speziell aufbereiteten schriftlichen Unterrichtsmaterialien sowie durch Bearbeiten von schriftlich gestellten Aufgaben überwiegend theoretisches Wissen an (Fakten, Begriffe, Modelle etc.). Lernende kommunizieren über ein Medium mit dem Lehrenden, da ihre Standorte unterschiedlich sind. Der Lerner ist für die Aufnahme der Lerninhalte teilweise eigenverantwortlich und bestimmt Zeitpunkt, Zeitdauer und die Strategie der Informationsaufnahme.

Tabelle 6: Phasen des Fernunterrichts

| Phase | Beschreibung |
|--------------------|--|
| Orientierungsphase | Lerner informieren sich über das Lernangebot, ihre Möglichkeiten und ihre Interessen |
| Rezeptionsphase | Lerner erhält Sendungen mit Informationsmaterial |
| Interaktionsphase | Anwendung und Übung des neu erworbene Wissens, Rücksendung der Aufgaben an Lehrende |
| Rückmeldungsphase | Korrektur der Aufgabenlösung, Wiederholung von Teilen des Unterrichtsmaterials |

Fernunterricht basiert dabei auf den didaktischen Prinzipien des Lernens in Einzelarbeit, des Lernens mit Medien, der aufgabenbezogene Rückmeldung.

Die Möglichkeiten des Technologieeinsatzes bei dieser Methode sind äußerst vielfältig. Neben der vereinfachten Distribution von Lehrmaterialien und der multimedialen Umsetzung von Studienbriefen ist insbesondere die Verbesserung der Kommunikationsmöglichkeiten von Bedeutung. Dabei können durch Einsatz synchroner wie asynchroner Kommunikationsformen die Rückmeldungszyklen wesentlich verkürzt werden. Des Weiteren wird die Kooperation und Kollaboration zwischen Lernenden ermöglicht und somit eine weitere methodische Vielfalt gefördert.

2.3.8.3 Frontalunterricht (*Klassenunterricht, Vorlesung*)

Der klassische *Frontalunterricht* ist auch heute noch die primäre Methode in der universitären Lehre. Lehrer gesteuerte Gespräche, die durch Anschauungsmittel unterstützt werden, stehen im Mittelpunkt. Diese Unterrichtsform dient vor allem der Vermittlung fachspezifi-

schen Orientierungswissens. Diese Methode ist lehrerzentriert, es wird vorausgesetzt, dass der Lehrende in angemessener Form auf seine Umwelt und die Lernenden reagiert. Es sind zwar Fragen und Interaktionen möglich, die jedoch meist nur einen geringen Zeitanteil einnehmen und größtenteils im vorhinein geplant sind. Die didaktischen Prinzipien sind lehrergesteuertes Lernen, Lernen im Klassenverband und thematisch orientiertes Lernen.

Tabelle 7: Phasen des Frontalunterrichts

| Phase | Beschreibung |
|--------------------|--|
| Orientierungsphase | Verbindung zwischen bereits vorhandenem Wissen und dem neuen Themengebiet |
| Rezeptionsphase | geordnete Vorstellung neuer Lerninhalte |
| Interaktionsphase | Diskussion und Einordnung der Lerninhalte in Wissenszusammenhang der Lernenden |
| Festigungsphase | Üben der Inhalte |
| Anwendungsphase | Übertragung der Inhalte auf neue Themengebiete |

Der Technologieeinsatz bei dieser Methode beschränkte sich ursprünglich auf die mediale Präsentationsunterstützung durch Videos, Animationen oder durch Folien. Noch heute bieten sogar manche Selbstlernumgebungen nur eine Abfolge von Folien, die über das WWW verfügbar gemacht werden, [vgl. Schu2001]. Der Forschungsbereich des Teleteaching versucht, den Technologieeinsatz entscheidend zu verbessern [Klei1998, BBL2000]. Dabei werden klassische Vorlesungen durch Whiteboards und Präsentationsmedien unterstützt und dann übertragen. Aufzeichnungen stehen in der Regel auch asynchron zur Verfügung (Lecture on Demand).

Die Erfolgswahrscheinlichkeit dieser Art von Lernszenarien ist jedoch als gering einzustufen [vgl. Schul2001]. Der Einsatz innovativer Technologien verbessert nicht die Eigenschaft klassischer Vorlesungen, dass diese Methoden lehrerzentriert und kaum interaktiv ablaufen. Es zeigt sich, dass häufig die reine Virtualisierung von klassischen didaktischen Methoden nicht immer zum Erfolg führt.

2.3.8.4 Individueller Lernplatz (Lernecke, Selbstlernplatz)

Eine Methode, die stark mit den ersten CBT-Programmen als Einzelplatzlösung verbunden ist, ist der *Individuelle Lernplatz*. Lerner eignen sich mithilfe von ausgewählten und systematisch geordneten Texten und AV-Medien selbstständig Begriffs- und Faktenwissen an, das zu vorher erarbeiteten Fragestellungen in Beziehung steht. Es wird in einer Lernumwelt gelernt, die die Realität durch ikonische und symbolische Repräsentation abbildet, d. h. durch Texte und verschiedene Medien. Ein wesentliches Element besteht darin, dass das gespeicherte Wissen geordnet ist und in Beziehung zu definierten Aufgabenfeldern steht. Prinzipiell kann jeder Arbeitsplatz durch die Bereitstellung von Lernumgebungen zu einem individuellen Lernplatz werden, wobei häufig die Tatsache vernachlässigt wird, dass die räumliche Trennung von Arbeits- und Lernplatz ein wichtiger Motivationsfaktor ist.

Didaktisches Prinzip ist das selbstgesteuerte Lernen, wobei die Steuerung der Lernprozesse entweder durch den Lernenden selbst oder durch die Lernumgebung unterstützt wird. Ferner sollte ein enger Bezug zwischen der Erkennungsstruktur des Lerners und den zugänglichen Wissensordnungen bestehen.

Tabelle 8: Phasen des Individuellen Lernplatzes

| Phase | Beschreibung |
|-------------------|---|
| Einrichtungsphase | Bereitstellung der Lerninhalte und Hilfsfunktionen für den Lernplatz |
| Klärungsphase | Abklärung der Lerninhalte und –präferenzen, Lernbeginn |
| Lernphase | Auswahl des Lernweges, Durchführung des selbstgesteuerten Lernprozesses |
| Bewertungsphase | Individuelle Überprüfung der Lerninhalte |

Diese didaktische Methode entstand durch neue Technologien und eine wachsende Popularität (insbesondere CBT) selbstgesteuerter und -verantwortlicher Lernprozesse, daher war der Technologieeinsatz integraler Bestandteil der Methode. Es ist offensichtlich, dass durch unterschiedliche Hilfe- und Unterstützungsfunktionen bzw. unterschiedliche Qualität der Medien starke Unterschiede der konkreten Umsetzung auftreten können.

2.3.8.5 Individualisierter Programmierter Unterricht, Programmierte Unterweisung

Lerner eignen sich mithilfe programmierter Lehrtexte in kleinen Lernschritten selbstständig und individuell Kenntnisse und Fertigkeiten an, die genau festgelegt sind. Komplexe Prozesse oder Vorgänge werden dabei in zeitlicher und logischer Abfolgen gegliedert. Eine Lernschrittfolge soll bis zum Lernziel führen. Diese behavioristische Methode basiert auf dem Prinzip der Konditionierung. Durch Verstärkungs- und Rückmeldungsmechanismen wird erwartet, dass sukzessive das Lernziel erreicht wird. Als didaktische Prinzipien sind individualisiertes Lernen und zielerreichendes Lernen (mastery learning) zu nennen.

Tabelle 9: Phasen der Programmierten Unterweisung

| Phase | Beschreibung |
|--------------------|--|
| Vorbereitungsphase | Entwicklung des Lernprogramms |
| Rezeptionsphase | Überprüfung der Eingangsvoraussetzungen, Einweisung |
| Interaktionsphase | Erarbeiten der Lerninhalte |
| Rückmeldungsphase | Lernzielkontrolle durch Zwischen- und Abschlusstests |

Diese Methode wurde gerade in der ersten Entwicklungsphase computerunterstützter Lernumgebungen angewendet. Die Aufbereitung der jeweils kurzen Lerneinheiten durch entsprechende Medien ging mit dieser Entwicklung einher. Insbesondere durch Anpassung an Lernende durch tutorielle Funktionen wurde diese Methodik erweitert.

2.3.8.6 Simulations- und Rollenspiele

[Geut1989] beschreibt Planspiele als eine spezifische Tätigkeit, in der zahlreiche Spielteilnehmer, die sich zu mehreren Gruppen zusammenschließen, in bestimmten Rollen, wechselnden Szenen und Situationen interagieren, und zwar innerhalb einer hypothetisch-fiktiven Umwelt, die auf bloßen Annahmen beruht und dennoch möglichst realistisch erscheinen soll. Vielfach ist bereits in den Materialien zur Ausgangslage ein bestimmtes zentrales Problem vorgegeben. Die Spielteilnehmer haben die Aufgabe, ein zumeist höchst reales Handlungs- und Entscheidungsproblem in mehreren Spielrunden auf Lösungsmöglichkeiten hin durchzuspielen.

Eine eher auf den Kontext der Wirtschaftswissenschaften bezogene Beschreibung liefert [Grob1997]. Er definiert Planspiele als „eine Lehr-/Lernmethode, bei der fiktive Probleme systematisch erzeugt und im Team gehandhabt werden. Die Prozessarchitektur eines Planspiels beinhaltet die Spielbeschreibung und das Planspielmodell, das eine vereinfachte Abbildung eines Ausschnitts der Realität darstellt. Das Planspielmodell repräsentiert dabei nicht nur den Abrechnungsmodus (z. B. in Form einer Bilanz und einer Gewinn- und Verlustrechnung), sondern auch die Wettbewerbssituation, in der sich die Planspielgruppen befinden. Der Planspielprozess kann in manueller Form oder computerunterstützt ablaufen. Die Leitung einer Planspielsitzung obliegt einem Moderator, der neben organisatorischen Funktionen, wie z. B. der Gruppenbildung, vor allem die fachliche Betreuung der Teilnehmer innehat. Zu seinen Aufgaben gehören dabei nicht nur die Erläuterung und Interpretation von Spielregeln, sondern auch die pädagogisch geschickte Verknüpfung von Planspielinhalten und traditionell vermitteltem Lehrstoff.“ Zusammenfassend lassen sich folgende Kernbereiche eines Planspiels identifizieren:

- Realitätsnahes Simulationsmodell
- Konkurrenzgedanke
- Planen von Entscheidungen
- Bedeutung und Aufgaben des Moderators

Die didaktischen Prinzipien beruhen auf explorativem Lernen, Lernen durch Erfahrung und Modelllernen. Die Phasen eines Simulationsspiels können wie folgt beschrieben werden. Es ist zu beachten, dass die Aktivitäts- und Reflexionsphase mehrfach durchlaufen werden, um konkrete Erfahrungen zu abstrahieren und daraufhin wieder anzuwenden.

Tabelle 10: Phasen des Simulationsspiels

| Phase | Beschreibung |
|------------------|--|
| Motivationsphase | Erläuterung der Spielsituation, Methodenanleitung |
| Aktivitätsphase | Spieldurchführung, Konkrete Erfahrungen, Experimentieren |
| Reflexionsphase | Diskussion der Ergebnisse, Abstraktion der Erfahrungen |
| Evaluationsphase | Auswertung und Zusammenfassung |

Der Computereinsatz bei dieser Methode ist naheliegend, da Simulationen in verschiedenen Kontexten Erklärungsansätze für komplexe Sachverhalte oder eine Visualisierung des Systemverhaltens liefern. Basierend auf einem Simulationsmodell können Entscheidungen getroffen und deren Auswirkungen entdeckt werden. Daher bietet diese Methode eine Unterstützung für komplexe Lernziele [vgl. AdBP2000b, ABKP1999, Brem2000].

2.3.8.7 Kritische Würdigung

Eine vielfach angewandte Vorgehensweise bei der didaktischen Planung von Lernumgebungen ist der Einsatz von Technologien zur Umsetzung traditioneller didaktischer Methoden, wie es am Beispiel der Vorlesung gezeigt wurde. Diese Vorgehensweise ist nicht ohne genauere Betrachtung positiv zu beurteilen. Häufig wird ein hoher technologischer Aufwand zur Umsetzung betrieben, der jedoch in keinem Verhältnis zum Nutzen steht. Bezeichnend sind die Evaluationsergebnisse verschiedenster Studien zum Nutzen computerunterstützter Lernumgebungen bei [Russ1999, Russ2001]. Das *No-Significant-Difference-Phänomen* zeigt, dass nicht alle aufwendigen Konzeptionen notwendigerweise zu positiven Ergebnissen führen. Daher ist es notwendig, jede didaktische Konzeption permanent zu evaluieren und entsprechende Folgerungen in eine neue Planung einzubeziehen. Aufgrund der Komplexität der Einflussfaktoren ist keine Empfehlung für eine bestimmte didaktische Form präskriptiv vorzuschreiben, es können nur Empfehlungen ausgesprochen werden, die dann an den Kontext angepasst werden müssen. [Schul2001] beschreibt eine solche Empfehlung als *acht Imperative für Lernumgebungen*, für die computerunterstützte Lernumgebungen mit hoher Wahrscheinlichkeit positive Effekte liefern werden:

- *Eignung der Lerninhalte*: Nicht alle Lerninhalte können vollständig durch virtuelle Lernsituationen substituiert werden, wie z. B. Exkursionen oder Coaching-Sitzungen. Als weiteres Beispiel für eine Lernsituation, die eine Präsenzphase erfordert, wird in Abschnitt 6.2.2 die Gruppenbildungsphase eines virtuellen Weiterbildungsstudienganges beschrieben.
- *Didaktische Angemessenheit*: [Schu2001] zählt als Negativbeispiele insbesondere textbasierte Lernumgebungen oder Umsetzungen von Powerpoint-Präsentationen auf, während sich z. B. Planspiele oder Visualisierungen in Mathematik oder Informatik besonders zur Umsetzung eignen.
- *Hypertextbasierte Umsetzung*: Lerninhalte sollten nicht in linearer, sondern in vernetzter Form präsentiert werden.
- *Selbstgesteuertes Lernen*: Um die Flexibilität von Lernumgebungen zu erhöhen, sollte ein Fokus auf selbstgesteuertes Lernen gelegt werden. Dabei ist auf interaktive Elemente zu achten, die insbesondere den konstruierenden Umgang mit Lernressourcen enthalten sollten.
- *Authentische Lerninhalte*: Die Lernumgebungen sollten authentische und damit komplexe Lerninhalte enthalten und damit zu problemorientierten Lernansätzen führen.
- *Lernerzentrierung*: Ausrichtung des Lernens auf die Lernenden.
- *Bildung von Lerngemeinschaften*: Die Bildung von Lerngemeinschaften integriert kommunikative und soziale Elemente. Gerade in Reflexionsphasen kann durch die Nutzung der Kommunikationsanwendungen ein Diskurs angestoßen werden.

- *Anspruchsvolle Prüfungsformen:* Einfache Testverfahren wie Multiple Choice werden der Überprüfung komplexer Inhalte nicht gerecht und lassen nicht auf den Lernerfolg schließen. Daher müssen adäquate Übungen und Prüfungsformen entwickelt werden.

Abschließend ist die Notwendigkeit festzustellen, Evaluationsergebnisse langfristig und organisationsübergreifend zur Verfügung zu stellen, zu vergleichen und interorganisational zu nutzen. Daher wird eine einheitliche Darstellungsform benötigt, die Zusammenhänge von Lernziel, Lernsituation, Methodik, Kommunikation, Präsentation und Evaluation spezifiziert. Damit wird die Vergleichbarkeit von Evaluationsergebnissen ermöglicht.

2.3.9 Evaluation

Der Begriff der Evaluation muss wie schon der Begriff der Didaktik jeweils in unterschiedlichen Anwendungsfeldern und aus unterschiedlichen Perspektiven betrachtet werden. So kann sich Evaluation zum Beispiel auf Lehrende, Entwickler oder gesamte Systeme beziehen. Ebenso kann sich die Evaluation auf die Leistung der Lernenden beziehen, d. h., es wird überprüft, welche Lernerfolge in einer Lernumgebung erzielt wurden. An dieser Stelle werden zunächst verschiedene Evaluationsverfahren dargestellt und am Beispiel der Lernerfolgsüberprüfung erläutert. Im weiteren Verlauf der Arbeit (Abschnitt 5.1) wird zusätzlich die Evaluation der Entwicklungs- und Lernprozesse als Teilbereich einer integrierten Qualitätssicherung diskutiert.

[Kerr2001] betrachtet in diesem Zusammenhang zunächst den Begriff des Lernerfolgs. Die Überprüfung reinen Faktenwissens und deren Abfrage in Form von einfachen Testverfahren bilden die Grundlage von Methoden wie der Programmierten Unterweisung. Offensichtlich werden bei komplexeren Aufgabenstellungen auch die Überprüfungsmethoden komplexer und sind nicht mehr in einfachen Verfahren, wie Multiple Choice, zu überprüfen. Des Weiteren werden bei derartigen Testverfahren Kriterien wie Persönlichkeitsbildung oder Aufbau und Infragestellung kognitiver Schemata nicht einbezogen [Kerr2001]. Während aus kognitivistischer Sicht die Objektivierbarkeit und Überprüfbarkeit von Wissen gegeben ist, kann im Sinne des Konstruktivismus höchstens untersucht werden, inwieweit sich gemeinsame Sichtweisen und Perspektiven auf ein Wissensgebiet gebildet haben. Es kann insgesamt also nur betrachtet werden, ob Lernumgebungen „soziale settings beeinflussen und die Wissenskonstruktion in diesen settings unterstützen“ [Kerr2001].

Trotz dieser generellen Problematik der Lernerfolgsüberprüfung ist gerade im Hochschulkontext eine formalisierte Überprüfung der Studienleistung weiterhin notwendig und muss im didaktischen Design berücksichtigt werden. Daher sollen kurz verschiedene Möglichkeiten und Testverfahren sowie korrespondierende Möglichkeiten der Computerunterstützung aufgezeigt werden.

In Anlehnung an [Hein2000] kann eine Evaluation in folgende Phasen unterteilt werden:

1. *Festlegen der Evaluationsobjekte/Formulieren des Evaluationsziels:* Zunächst müssen das Ziel der Evaluation und deren Objekte festgelegt werden. Im Zusammenhang der Lernerfolgsevaluation werden die Leistungen der Lernenden überprüft, die Zielsetzung ist die Beherrschung verschiedener Lerninhalte.

2. *Ableiten und Gewichten der Evaluationskriterien:* In dieser Phase werden Kriterien und deren Gewichtung festgelegt. Dabei können zum Beispiel Lernziele und deren genauere Beschreibung bestimmt und priorisiert werden.
3. *Abbilden der Evaluationskriterien in Metriken:* Gerade bei komplexen Lernzielen ist die Abbildung in eine formale Metrik nicht ohne weiteres möglich. Es wird jedoch versucht, eine messbare Eigenschaft der Lernziele festzulegen.
4. *Auswählen der Messmethoden:* In dieser Phase werden Methoden bestimmt, um die Überprüfung durchzuführen. Im universitären Kontext bedeutet dies die Festlegung der Prüfungsform und deren Ausgestaltung.
5. *Durchführen der Messung:* In dieser Phase wird die Messung vorgenommen, z. B. die Durchführung einer Prüfung.
6. *Auswerten der Messdaten:* In der letzten Phase werden die Daten ausgewertet und interpretiert. Dabei werden z. B. bei Abschlussprüfungen Noten festgelegt und entsprechende Zertifizierungen vorgenommen. Bei studienbegleitenden Prüfungen werden des Weiteren entsprechende Maßnahmen ergriffen. So können zum Beispiel Lerninhalte oder -methoden angepasst werden. Diese Phase schließt somit nicht mit einer Beurteilung, sondern impliziert, dass jeweils entsprechende Maßnahmen ergriffen werden.

Im Bereich der Lerntechnologien wird insbesondere die Durchführungsphase durch Applikationen unterstützt. Dabei werden im Allgemeinen vier Klassen von Evaluationsverfahren unterschieden:

- Fragebögen
- Interviewverfahren
- Beobachtungstechniken
- und Testverfahren

Fragebögen stehen dabei häufig im Vordergrund. Gerade Lernmanagementsysteme bieten meist die einfache Generierung von Single/Multiple Choice-Tests und Zuordnungsaufgaben. Es stellt sich jedoch die Frage, inwieweit komplexe Lernziele durch diese Verfahren überprüft werden können. Nur durch eine umfassende Kombination aus den genannten Verfahren können Lernleistungen annähernd bewertet werden.

2.3.10 Zusammenfassung

Es zeigt sich, dass die didaktische Konzeption und Durchführung eine Vielzahl an Einflussfaktoren beinhaltet und nicht ohne weiteres verallgemeinert werden kann. Die zuvor diskutierten Faktoren werden in der Entwicklung des Essener-Lern-Modells aufgegriffen. Der Entwicklungsprozess von Lernumgebungen wird hier durch entsprechende Funktionen zu deren Spezifikation unterstützt.

Abschließend werden diese Faktoren in Tabelle 11 zusammengefasst, um eine Übersicht über maßgebliche Determinanten zur Verfügung zu stellen.

Tabelle 11: Determinanten der Entwicklung von Lernumgebungen

| Determinante | Ausprägungen |
|------------------------|---|
| Lerntheorie | Behaviorismus, Kognitivismus, Konstruktivismus |
| Lernziele, Dimensionen | <i>kognitiv</i> : Wiedergeben, Verstehen, Anwenden, Analysieren, Synthetisieren und Beurteilen <i>affektiv</i> : Aufnehmen, Werten, Reagieren, Werteordnung, Bestimmtsein durch Werte <i>psychomotorisch</i> : Imitation, Manipulation, Präzision, Handlungsgliederung, Naturalisierung <i>sozial</i> : Teamfähigkeit, Konfliktmanagement, Durchsetzungsvermögen |
| Lernziele, Abstraktion | Richtziele, Grobziele, Feinziele |
| Lerninhalte | Fakten/kontextfreie Regeln, kontextabhängige Regeln, Problemlösung, komplexe Situationen, Gestalt-/Mustererkennung |
| Rollen | Lernender, Lehrender, Entwickler, Manager, Domänenexperte, Mediendesigner |
| Lernzeitpunkt | synchron, asynchron, Mischformen |
| Lerndauer | geschlossen, offen |
| Raum | gleich, verteilt, Mischformen |
| Interaktionsgrad | Kommunikation zwischen Lehrendem, Lernenden, Computer |
| Benutzergruppe | Vorwissen, Skills, Lernpräferenzen, situativer Kontext |

2.4 Architektur computerunterstützter Lernumgebungen

Unter einer Architektur versteht man die Struktur eines Informations- und Kommunikationssystems, insbesondere die Komponenten und deren Beziehungszusammenhänge [HeRo1989]. Dies bezieht die Art, funktionale Eigenschaften und ihr Zusammenwirken ein [Sche1998b]. Dabei können Ziel und Detailliertheitsgrad einer Architektur unterschiedlich sein. Einerseits kann die Architektur eine Übersicht über ein Informationssystem geben, andererseits kann sie eine genaue Beschreibung der Komponenten liefern. Zunächst wird die Zielsetzung der Architektur verdeutlicht:

- *Darstellung der Systemkomponenten und Interdependenzen:* Die grundlegenden Komponenten von Lernumgebungen und deren Zusammenhänge sollen verdeutlicht werden. Dabei müssen die in Abschnitt 2.3 spezifizierten Beschreibungsmerkmale abzubilden sein.
- *Einordnung von Lernumgebungen:* Die Architektur soll die Einordnung eines beliebigen Systems ermöglichen, um Lernumgebungen vergleichbar zu machen. Dabei ist die Konsistenz zu den in Abschnitt 2.3 dargestellten Einflussfaktoren sicherzustellen.
- *Entscheidungshilfe:* Die Architektur soll eine Übersicht über die Gestaltungsmöglichkeiten von Lernumgebungen aufzeigen und somit als Unterstützung für Akteure dienen.
- *Generische Architektur:* Es soll eine generische Architektur entwickelt werden, die die Einordnung aller möglichen Instanzen erfasst.

Unter Beachtung dieser Zielsetzungen werden im Folgenden ausgewählte Architekturen beschrieben und auf ihre Verwendbarkeit im Rahmen des Essener-Lern-Modells hin untersucht. Dadurch werden generelle Komponenten und mögliche Instanzen dargestellt und deren Zusammenhänge erläutert.

2.4.1 Architekturen von Lernumgebungen

Im Folgendem werden exemplarisch Architekturen betrachtet, die unterschiedliche Sichtweisen und Detailliertheitsgrade aufweisen. Es wird in diesem Abschnitt analysiert, welche Architekturen in einem Vorgehensmodell zur Entwicklung von Lernumgebungen verwendet werden können.

[Fick1992] beschreibt eine Architektur, die die verschiedenen Komponenten von Lernsystemen schematisch darstellt:

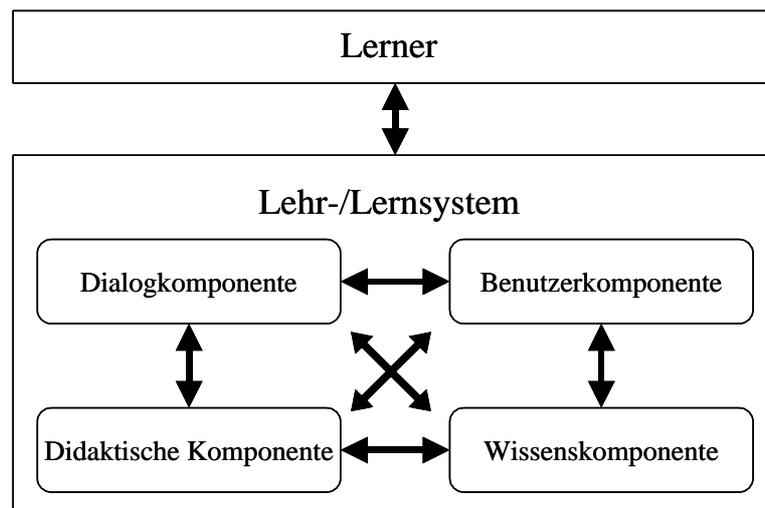


Abbildung 15: Architektur nach [Fick1992]

Das *Lehr-/Lernsystem* besteht aus vier Komponenten: Die *Wissenskomponente* enthält ein Modell des zu vermittelnden Wissens und der Lerninhalte. Dabei ist zu betrachten, welche Art der Wissensrepräsentation gewählt wird. Eine *Benutzerkomponente* enthält Informationen über die Lernenden, die in Form von Stereotypen repräsentiert werden. Eine *didaktische Komponente* ist verantwortlich für die Umsetzung von Lehrstrategien, während die *Dialogkomponente* eine Schnittstelle für den Lernenden bietet.

Diese Architektur enthält die Basiskomponenten von Lernumgebungen und kann als generelle Übersicht zur Einordnung von Lernumgebungen dienen. Dabei werden jedoch im Modell keine Instanzierungen betrachtet. Des Weiteren fehlt eine Darstellung der Prozesse innerhalb des Lernprozesses, da diese nur implizit in der didaktischen Komponente enthalten sind. Ebenso werden keine weiteren Akteure wie z. B. Lehrende einbezogen. Daher kann diese Architektur nur als Übersicht, nicht aber als Implementierungsrichtlinie genutzt werden.

Einen umfassenderen Ansatz verfolgen [FHSU2000, FeSc2001] bei der Darstellung einer Gesamtarchitektur Integrierter Internet-Lernumgebungen.

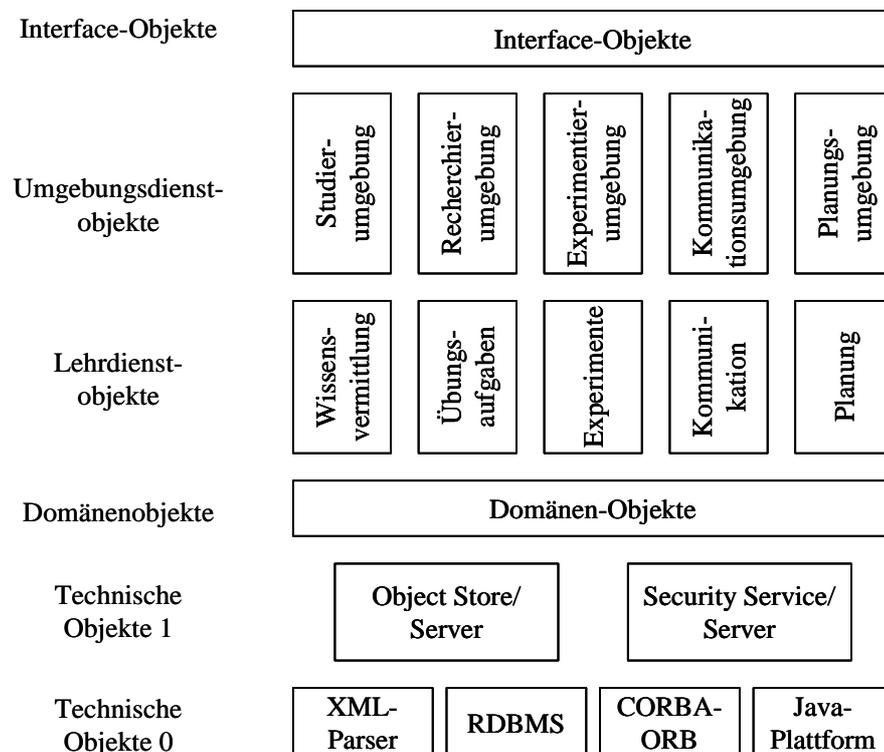


Abbildung 16: Architektur Integrierter Internet-Lernumgebungen [FeSc2001]

Aufgrund der Verwendung generischer Objekte (Interface-Objekte, Umgebungsdienstobjekte, Lehrdienstobjekte, Domänenobjekte, Technische Objekte) ist die Architektur grundsätzlich zur Beschreibung der Struktur von Lernumgebungen sinnvoll einsetzbar. Der Beschreibungsumfang ist als hoch einzustufen, da sowohl didaktische Komponenten (Dienstobjekte) als auch technische Komponenten enthalten sind. Insbesondere die technischen Objekte geben eine detaillierte Richtlinie, in welcher Form internetbasierte Lernumgebungen implementiert werden können. Da diese Architektur jedoch zur Beschreibung eines

spezifischen Systems verwendet wurde [FeSc2001], müssen die einzelnen Objekte generalisiert werden, um als allgemeingültige Architektur verwendet werden zu können. Insbesondere die technischen Objekte müssen generalisiert werden, um weitere Implementierungsmöglichkeiten zu umfassen. Ebenfalls werden bei den Dienstobjekten bereits Annahmen über die didaktische Gestaltung getroffen, so dass nicht jede didaktische Strategie darstellbar ist.

Die Architektur von Lernsystemen nach [Paqu2001] betrachtet insbesondere die Entwicklungssicht und bezieht maßgebliche Prozesse ein (Engineering, Entwicklung, Wissensassoziation, Übermittlung).

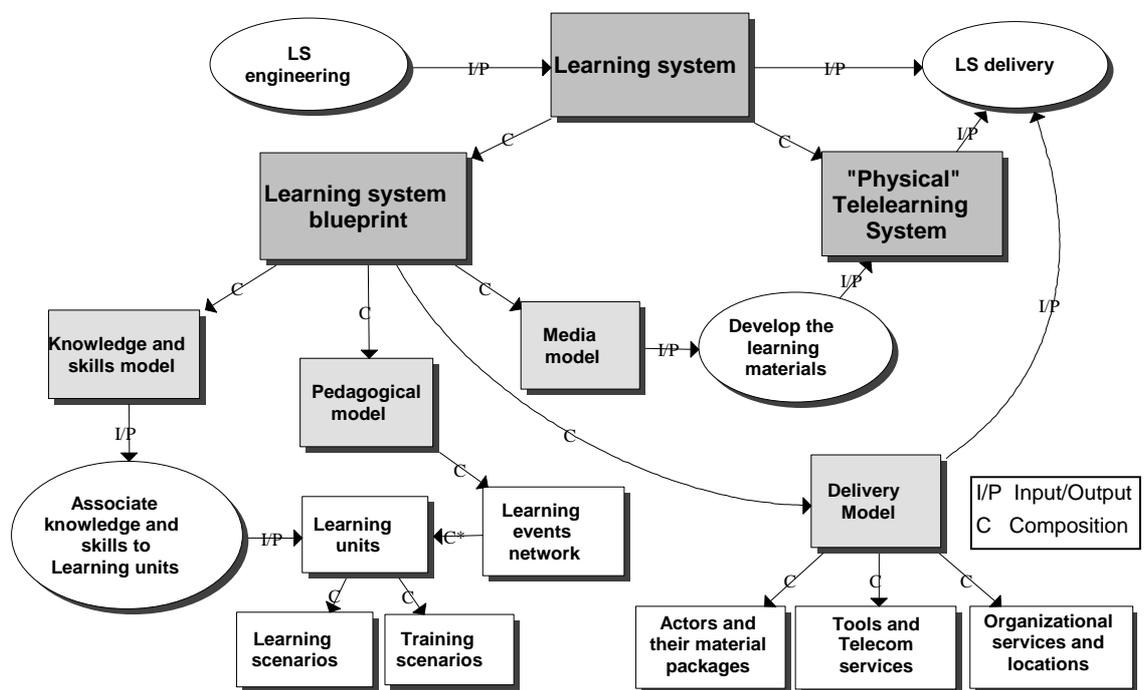


Abbildung 17: Architektur von Lernumgebungen nach [Paqu2001]

Diese Architektur bildet die maßgeblichen Prozesse und Komponenten zur Entwicklung von Lernumgebungen ab (Abbildung 17). Durch die Zuordnung möglicher Instanzen können Lernumgebungen beschrieben und eingeordnet werden. In diesem Modell ist jedoch die Benutzerkomponente nur gering ausgeprägt und muss daher ebenfalls erweitert werden. Zusammenfassend dient diese Architektur als Richtlinie zur Entwicklung und Erstellung von Lernumgebungen und kann als Basis zur Beschreibung von Lernumgebungen und Entwicklungsprozessen verwendet werden.

2.4.2 Learning Technology Systems Architecture (LTSA)

Die LTSA beschreibt ein abstraktes Modell der Architekturen von Lernumgebungen und wurde vom IEEE Learning Technology Standards Committee (LTSC) entwickelt. Aufgrund der thematischen Zusammengehörigkeit wird dieser Ansatz, der als Vorschlag für eine Referenzarchitektur als ISO Standard vorliegt, schon in diesem Abschnitt behandelt. Weitergehende Erläuterungen zu Lerntechnologiestandards folgen in Kapitel 4.

Die Architekturspezifikation stellt eine grundlegende Struktur bereit, um existierende und zukünftige Lernsysteme zu verstehen, zu analysieren und zu vergleichen. Die LTSA-Spezifikation ist unabhängig vom jeweiligen pädagogischen Konzept sowie soziokulturellen Kontext, ferner ist sie inhalts- und plattformunabhängig. Die Einteilung in verschiedene Layer beschreibt die Geltungsbereiche der einzelnen Architekturebenen (Abbildung 18).

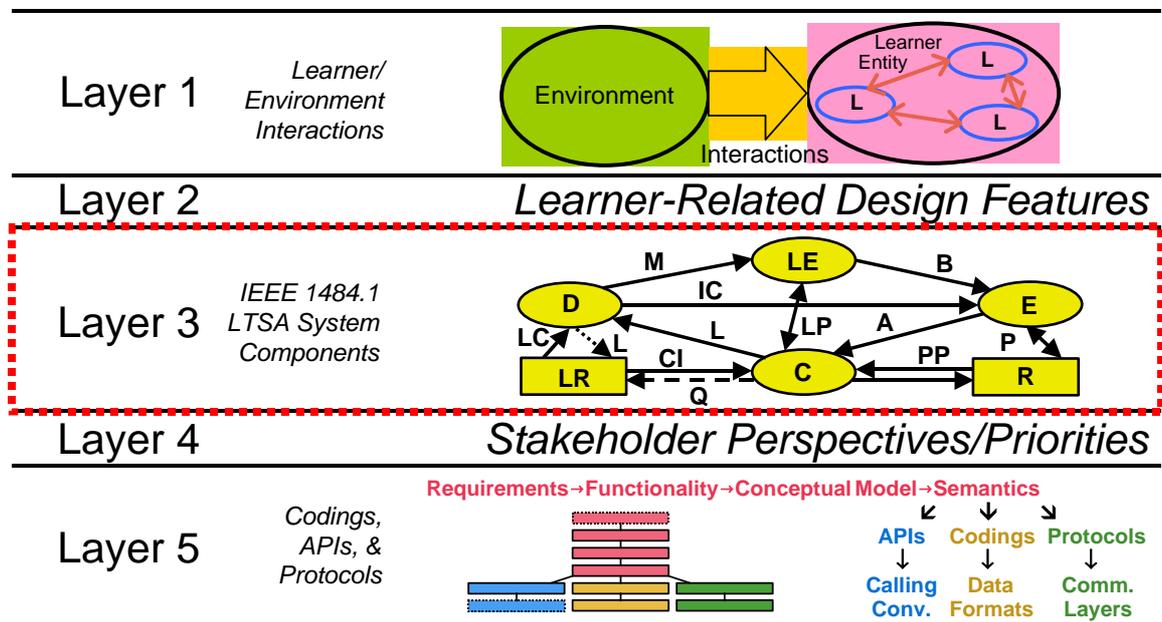


Abbildung 18: LTSA-Ebenen [LTSC2001a]

Die LTSA unterteilt sich in fünf Ebenen, wobei nur die dritte Ebene normativen Charakter hat – alle anderen Ebenen werden als informativ bezeichnet und bilden nur einen generellen Rahmen für die Architektur. Die erste Ebene befasst sich mit dem Erwerb, Transfer und Austausch von Wissen des Lernenden mit seiner Umwelt. Dabei ist dieser Level bis auf Erläuterungen nicht näher spezifiziert. Es werden nur mögliche Objekte der Umwelt aufgelistet (Lehrender, Medien, Eltern etc.), die einen Einfluss auf den Lernenden haben können. Auf der zweiten Ebene werden Einflüsse des Lernenden auf die Lerntechnologie betrachtet. Diese Faktoren liegen laut LTSA außerhalb des Geltungsbereichs des Standards. Die maßgebliche Ebene dieses Standards ist die dritte Ebene, auf der Prozesse und Akteure spezifiziert werden, die an Lernprozessen beteiligt sind. Auf der vierten Ebene werden dann mögliche Subsysteme der Architektur betrachtet, die mögliche Implementierungen beschreiben, während auf der fünften Ebene interoperable Kodierungen, APIs und Protokolle spezifiziert werden, die zur Implementierung verwendet werden können.

Wie bereits erwähnt, wird der Geltungsbereich des Standards auf die dritte Ebene, die Systemkomponenten, beschränkt.

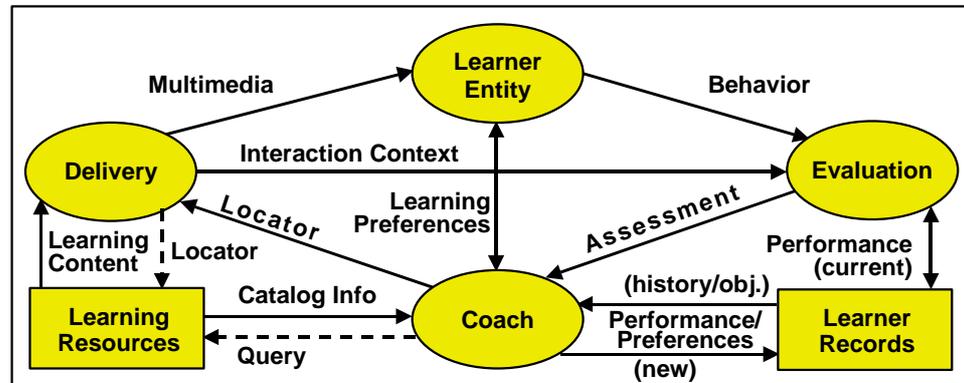


Abbildung 19: LTSA-Systemkomponenten [LTSC2001a]

Der allgemeingültige Charakter der Spezifikation solcher Systeme und seiner Komponenten ermöglicht eine Systemarchitektur, die wiederverwendbar und anpassungsfähig ist. Neben der standardisierten Darstellung der *Prozesse* (Processes), des *Datenaustausches* (Flows) und der *Speicher* (Stores) werden Formate spezifiziert, um die Kommunikation und Integration verschiedener Systeme zu ermöglichen. Dadurch können u.a. die Entwicklungszeiten und daraus folgend die Produktionskosten bei der Implementierung von Lernumgebungen reduziert werden. Die Spezifikation befindet sich zurzeit noch in der Entwicklung (Draft 8), d. h., es handelt sich hierbei um einen nicht verabschiedeten IEEE Standard. Abbildung 19 zeigt die LTSA im Überblick.

- Ein *Process* beschreibt die aktiven Systemkomponenten/Teilsysteme einer Lernumgebung, die ihre Eingaben in Ausgaben (Flows) umwandeln: *Learner Entity*, *Evaluation*, *Coach* und *Delivery*.
- *Flows* beschreiben den Informationsfluss zwischen den Komponenten, d. h., Ein- und Ausgaben der Stores und Processes: *Learning Preferences*, *Behavior*, *Assessment Information*, *Performance Information*, *Query*, *Catalog Info*, *Learning Content* und *Interaction Context*.
- Die Funktion von *Stores* ist die Speicherung von Informationen: *Learning Resources*, *Learner Records*.

Ein *Learner Entity* repräsentiert einen einzelnen Lernenden oder eine Gruppe von Lernenden, wobei diese individuell oder kollaborativ arbeiten können. Das *Learner Entity* wird bei seinen *Multimedia* Interaktionen beobachtet und beurteilt (*Behavior*). Der Evaluierungsprozess liefert Einschätzungen und/oder Leistungsinformationen (*Assessment Information*), ferner werden diese Informationen in den *Learner Records* gespeichert. Die Evaluation wird durch den *Interaction Context* unterstützt, da für eine sorgfältige Beurteilung sowohl das endogene als auch das exogene Umfeld der Lernumgebung von entscheidender Bedeutung sind. Der *Coach* koordiniert verschiedene Systemkomponenten. So werden beispielsweise die jeweiligen *Learning Preferences* zwischen ihm, dem *Learner Entity* und anderen Stakeholdern (z. B. Eltern oder Mentoren) vereinbart. Dieser Vorgang wird durch

verschiedene Aspekte, wie soziokulturelle Einflüsse, beeinflusst. Ferner überprüft der Coach die *Assessment* und *Performance Information* des Learner Entity. Anhand dieser Informationen wird der Auswahlprozess für weitere Lerninhalte unterstützt. Aufbauend auf diesen Informationen sendet der Coach eine Anfrage (*Query*) an die *Learning Resources*, um nach entsprechenden Lernmaterialien zu suchen. Als Antwort erhält er eine *Catalog Info*, eine Liste von *Locators*, wie z. B. Uniform Resource Locators (URL). Die verschiedenen Locators werden vom *Delivery* Prozess verwendet, um über den entsprechenden *Learning Content* verfügen zu können. Die Lernmaterialien werden dann in multimedialer Form dem Learner Entity präsentiert.

Die LTSA ist eine abstrakte Architektur, die einen Rahmen für weitere Standardspezifikationen und die Einordnung von Lernsystemen bietet. Dennoch ist die Nutzung als Referenzmodell in ihrer derzeitigen Form nicht ohne weiteres möglich. Es müssen Implementierungsrichtlinien und Ausprägungen für die einzelnen Architekturkomponenten spezifiziert werden, um eine Einordnung zu ermöglichen und damit einen Nutzen für den Anwender zu schaffen.

Eine Alternative, die derzeit in der ISO/IEC JTC1 SC36, einem internationalen Standardisierungsgremium der ISO (siehe 4.2) für Lerntechnologien, diskutiert wird, ist die Architektur *Expertise and Role Identification for Learning Environments (ERILE)* [Lind2001a, Lind2001b]. Diese Architektur dient zur Beschreibung von Lerntechnologien. Der Schwerpunkt liegt, neben der Darstellung der Systemkomponenten, auf der Beschreibung der Akteure in Lern- und Entwicklungsprozessen und in der Darstellung der Expertise. Durch diese erweiterten Konzepte wird eine genauere Beschreibung von Lernprozessen ermöglicht. Die Ebene drei der LTSA wird somit um Aktorenkonzepte erweitert.

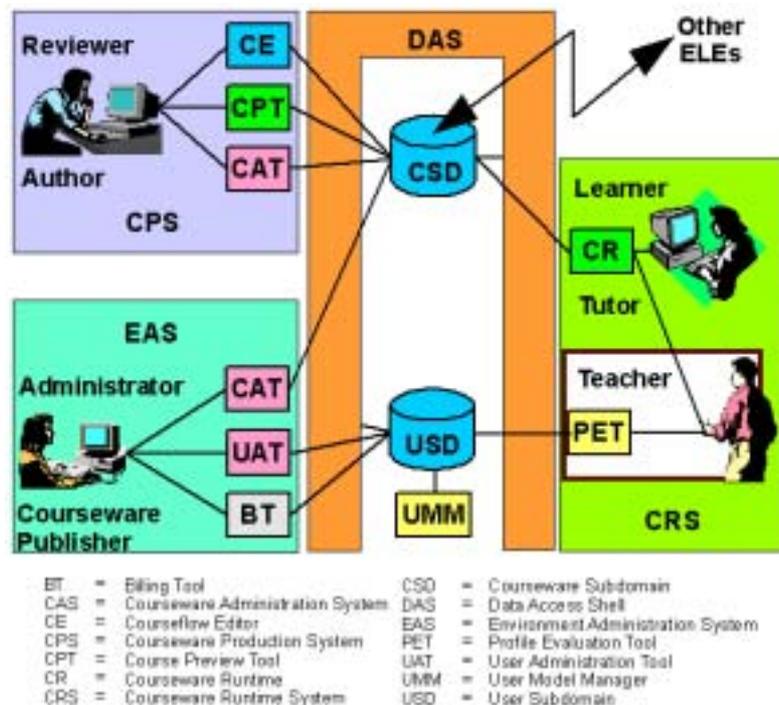


Abbildung 20: Architektur ERILE [Lind2001b]

2.4.3 Folgerungen

Betrachtet man die exemplarisch dargestellten Architekturen, so wird ersichtlich, dass verschiedene Schwerpunkte gesetzt werden und die Eignung der Architekturen nur für jeweils spezifische Zielsetzungen gegeben ist. Alle zuvor genannten Ansätze bieten eine Beschreibungsmöglichkeit zur Einordnung von Lernumgebungen. Dennoch werden verschiedene Beschreibungskriterien vernachlässigt. Insbesondere die Darstellung und Beschreibung didaktischer Elemente findet keine Berücksichtigung. Zwar enthält der Ansatz von [Fick1992] explizit eine didaktische Komponente, diese ist jedoch nicht weiter formalisiert und bietet keine Ansätze, konkrete Instanzen zu beschreiben.

Die weiteren Ansätze enthalten nur implizite didaktische Ansätze. Sie ermöglichen zwar eine technologieorientierte Beschreibung, die zur genauen Charakterisierung und Vergleichbarkeit jedoch nicht ausreicht. Des Weiteren erlaubt die Struktur teilweise nicht die Integration von Standards, die einen wichtigen Aspekt zur Entwicklung von Lernumgebungen darstellen (siehe Abschnitt 4.2). Daher wird für das Essener-Lern-Modell eine Architektur entwickelt, die diese Schwachpunkte behebt.

3 Vorgehensmodelle

In den letzten Jahrzehnten wurde eine Vielzahl von Vorgehens-, Prozess- und Phasenmodellen mit unterschiedlichen Zielsetzungen entwickelt. Gerade im Bereich des Software-Engineering werden immer wieder neue Ansätze entwickelt, um Entwicklungs-, Implementierungs- und Wartungsprozesse unter verschiedenen Bewertungskriterien zu optimieren. So ist die Vorgehensmodellierung thematisch wie historisch dem Bereich der Softwaretechnik (Software-Engineering) zuzuordnen. Es stellt sich jedoch die Frage, ob der Anwendungskontext der vorliegenden Ansätze verändert werden kann, ohne das Vorgehensmodell zu evaluieren und anzupassen. Die Charakteristika computerunterstützter Lernumgebungen müssen daher in die Evaluation einbezogen werden. Des Weiteren wurden eine Vielzahl spezieller Vorgehensmodelle zur Entwicklung computerunterstützter Lernumgebungen entwickelt. Aufgrund der Vielzahl von Instanzen computerunterstützter Lernumgebungen (siehe Abschnitt 2.2) ist zu überprüfen, welches Vorgehensmodell für welche Instanzen von Lernumgebungen verwendet werden kann.

Zunächst wird eine Begriffsabgrenzung vorgenommen, um die in dieser Arbeit verwendete Terminologie zu erörtern. Dies ist aufgrund uneinheitlicher Begriffsverwendung und -interpretation notwendig. Es zeigt sich, dass sich drei Ebenen von Modellen für einen Softwareentwicklungsprozess unterscheiden lassen: *Vorgehensstrategien*, *Vorgehensmodelle* und *Prozessmodelle*. Diese werden in den folgenden Abschnitten erläutert und mithilfe eines Bewertungskatalogs auf ihre Eignung für die Entwicklung computerunterstützter Lernumgebungen hin untersucht (siehe 3.2). Aus der Bewertung werden abschließend Anforderungen an ein Vorgehensmodell formuliert, die als Basis für die Entwicklung des Essener-Lern-Modells dienen. Im Anschluss daran werden verschiedene Vorgehensmodelle aus den Bereichen der Softwaretechnik, Didaktik und weiteren Disziplinen analysiert, um ihre Eignung zur Entwicklung von Lernumgebungen zu untersuchen.

3.1 Begriffsabgrenzung

Die Terminologie im Bereich der Vorgehensmodellierung differiert in den verschiedenen Forschungs- und Anwendungsbereichen deutlich. Dies ist insbesondere durch die unterschiedliche Herkunft der Ansätze begründet. Während eine Vielzahl der Modelle aus dem Bereich der Softwaretechnik stammen, wurden weitere Vorgehensmodelle für spezifische Anwendungskontexte (z. B. Projektmanagement, Konfigurationsmanagement) entwickelt. So wurden Begriffe eines spezifischen Anwendungsbereichs verallgemeinert und dann außerhalb dieses Kontextes missverständlich gebraucht.

Insbesondere aus dem Bereich der Softwaretechnik sind eine Vielzahl von Strategien, Methoden, Verfahren und Werkzeugen hervorgegangen, die den Softwareentwicklungsprozess unterstützen. Unter Softwaretechnik versteht man „[...] das Fachgebiet der Informatik, das sich mit der Bereitstellung und systematischen Verwendung von Methoden und Werkzeugen für die Herstellung und Anwendung von Softwareprodukten beschäftigt.“ [HKLR1984] Eine weitere häufig verwendete Definition betrachtet Software-Engineering

als „[...] das ingenieurmäßige Entwerfen, Herstellen und Implementieren von Software sowie die ingenieurwissenschaftliche Disziplin, die sich mit Methoden und Verfahren zur Lösung der damit verbundenen Problemstellungen befasst.“ [Balz1996] Obwohl diese Definition sowohl sprachlich als auch inhaltlich nicht exakt ist, kann aus ihr gefolgert werden, dass das Software-Engineering ein strukturiertes Vorgehen und Problemlösungswissen für den gesamten Softwareentwicklungsprozess modelliert.

Vorrangiges Ziel dieses Bereichs ist die Unterstützung des Softwareerstellungprozesses, der vier Teilgebiete umfasst: Softwareentwicklung, Softwaremanagement, Qualitätssicherung sowie Wartung und Pflege [Balz1996]. Dabei ist zu berücksichtigen, dass das Teilgebiet der Wartung und Pflege in neueren Modellen (siehe 3.3) in den Entwicklungsprozess integriert ist und nicht als eigenständiger Teilbereich aufgefasst wird. Im Folgenden werden auch die Begriffe der Softwareerstellung und Softwareentwicklung synonym verwendet, das heißt, dass der Begriff der Softwareentwicklung im umfassenden Sinne Management-, Qualitätssicherungs- und Wartungsprozesse einschließt.

Abschließend lässt sich der Bereich des Software-Engineering als technologieorientierter Forschungsbereich auffassen, der durch Methoden, Verfahren und Werkzeuge den Prozess der Softwareentwicklung unterstützt. Dabei umfasst die Softwareentwicklung sowohl den spezifischen Entwicklungsprozess als auch Management- und Qualitätssicherungsprozesse.

Die entwickelten Modelle unterscheiden sich jedoch bezüglich der Abstraktion, Spezialisierung und Kontextabhängigkeit. Aus Gründen der besseren Verständlichkeit wird zunächst der Themenbereich der Vorgehensmodellierung abgegrenzt, um dann spezifische Definitionen abzuleiten.

[FiBM1998] erläutern Vorgehensmodelle durch die Bestandteile des Begriffes: „Das *Vorgehen* bei der Entwicklung von betrieblichen Anwendungen, also der gesamte Entwicklungsprozess, wird auf Basis von Beschreibungen und Anleitungen durch Strukturierung aus verschiedenen Sichten als *Modell* abgebildet und somit transparent und planbar“.

[GrSe1996] beschreiben den Begriff Vorgehensmodell als ein „[...] ablauforganisatorisches Konzept der Software-Entwicklung [...], bei dem ein komplexer Prozess in klar definierte, überschaubare Einheiten gegliedert wird. Durch Vorgehensmodelle wird Handlungswissen für die Erstellung von Software in Form von Prinzipien, Methoden und Werkzeugen zur Verfügung gestellt. Aus diesem allgemeingültigen Wissen soll ein Software-Entwicklungsprozess für *konkrete Anwendungen* abgeleitet werden.“

Zentraler Bestandteil ist in beiden Ausführungen die Strukturierung der Entwicklungsprozesse. Dabei beziehen sich Vorgehensmodelle insbesondere auf die ablauforganisatorische Sicht. Implizit ist in den Definitionen enthalten, dass Anweisungen und Problemlösungswissen für die beteiligten Akteure zur Verfügung gestellt werden müssen und somit eine aufbauorganisatorische Strukturierung gegeben sein muss.

Der Grad der Abstraktion und der Spezialisierung kann jeweils unterschiedlich sein. So kann zum Beispiel ein Vorgehensmodell zur Softwareentwicklung existieren, das das generelle Vorgehen in Entwicklungsprozessen regelt, andererseits kann das Modell spezifische Anweisungen geben (z. B. Benennungskonventionen in Datenbanken). Es ist offensichtlich, dass diese Modelle einen unterschiedlichen Detaillierungsgrad haben. [FiBM1998] und [Brem1998] unterscheiden drei Ebenen:

„Eine *Vorgehensstrategie (VGS)* ist die Fokussierung des repräsentierten Wissens einer soziotechnischen Umgebung (Entwicklungsphilosophie, Software-Werkzeuge, Projektorganisation) bezüglich der Art und Weise, wie Software-Systeme gestaltet und betreut werden.“ [FiBM1998] Diese Abstraktionsebene stellt einzelne Teilbereiche und einen Rahmen zur Vorgehensweise dar. Konkrete Arbeitsabläufe werden auf dieser Ebene nicht vorgegeben.

„Das *Vorgehensmodell (VGM)* ist ein Muster zur Beschreibung eines Entwicklungsprozesses auf Basis eines Entwicklungsschemas.“ [FiBM1998] Dabei umfasst der Entwicklungsprozess die gesamte Lebensdauer eines Software-Systems vom Projektbeginn bis zur Außerbetriebnahme des Systems. Vorgehensmodelle sind somit als Instanz einer Vorgehensstrategie anzusehen. Synonym werden häufig die Begriffe Prozessmodell, Softwareentwicklungsmethode oder Phasenmodell verwendet. Eine weitere Unterscheidung auf dieser Ebene wird zwischen generischen Vorgehensmodellen sowie Vorgehensmodellen für spezifische Projekttypen getroffen. So ist zum Beispiel das V-Modell [BrDr1995] als generisches Vorgehensmodell für alle Entwicklungsprojekte des Bundes zu verwenden, während das Hypertext Development Model (HDM) (siehe Abschnitt 3.4.2.2) nur bei der Entwicklung von Hypermedia-Anwendungen verwendet werden kann und somit auf diesen Projekttyp beschränkt ist.

Ein *Projektmodell (PJM)* ist eine für ein spezifisches Projekt angepasste Instanzierung eines Vorgehensmodells. Dabei werden die im Vorgehensmodell generisch verwendeten Komponenten (Rollen, Ablauf, Werkzeuge) für ein konkretes Projekt festgelegt.

Der Zusammenhang der Abstraktionsebenen ist in Abbildung 21 zusammenfassend dargestellt.

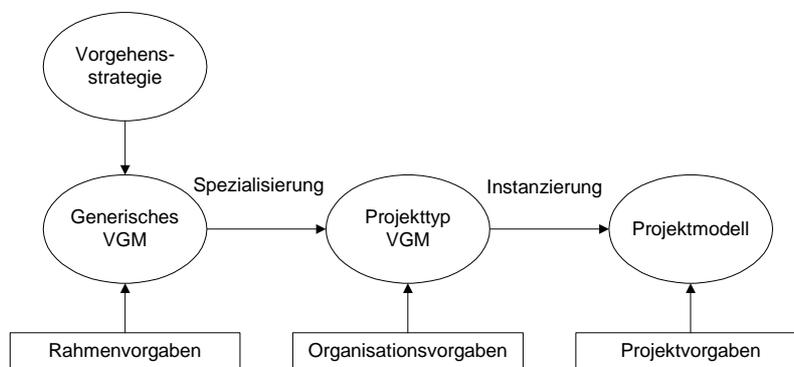


Abbildung 21: Beziehungen der Abstraktionsebenen

Daneben zeigt Abbildung 22 in Anlehnung an [FiBM1998] die Bestandteile eines Vorgehensmodells. Dieses Modell wird von der Fachgruppe *Vorgehensmodelle für betriebliche Anwendungssysteme* der Gesellschaft für Informatik e.V. verwendet und dient als Basis für die Entwicklung einer einheitlichen Terminologie im deutschen Sprachraum. Die Begrifflichkeiten in der vorliegenden Arbeit orientieren sich an diesem Ordnungsschema.

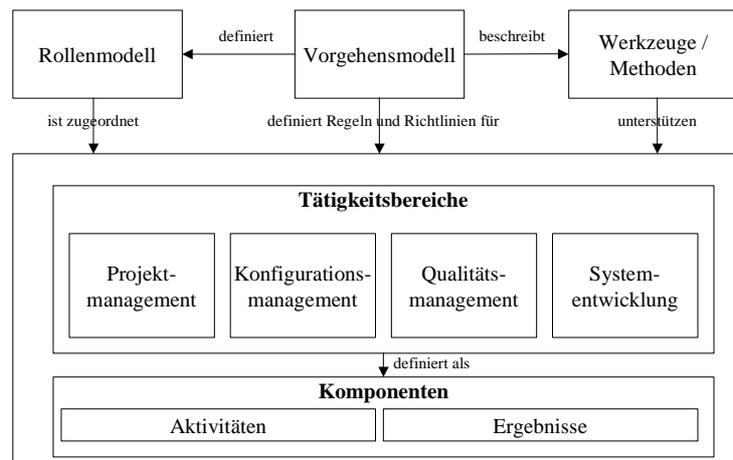


Abbildung 22: Ordnungsschema Vorgehensmodelle

Zentrale Bestandteile sind Definitionen, Regeln, Richtlinien und Empfehlungen, die den Ablauf eines Entwicklungsprojekts beschreiben. Die organisatorische Struktur wird in Form eines *Rollenmodells* abgebildet. Des Weiteren werden *Werkzeuge und Methoden* verwendet, Unterstützungsfunktionen für die Akteure des Entwicklungsprozesses bieten. Werkzeuge und Methoden umfassen auch interne und externe Richtlinien, wie Normen und Standards (siehe Kapitel 4), die im Entwicklungsprozess beachtet werden müssen. Ein Ablaufmodell, Rollen sowie Werkzeuge und Methoden werden verschiedenen Tätigkeitsbereichen zugeordnet:

- *Systementwicklung*: Dieser Teilbereich beschreibt die Aktivitäten und Ergebnisse, die unmittelbar mit der Erstellung einer betrieblichen Anwendung zusammenhängen, wie beispielsweise das Design oder die Implementierung einer Anwendung.
- *Projektmanagement*: Das Projektmanagement umfasst die Rahmenbedingungen eines Projekts, wie zum Beispiel die Formulierung von Projektzielen, Projektplanung, Projektsteuerung, Dokumentation und Controlling.
- „Unter *Konfigurationsmanagement* versteht man die Identifikation der Konfiguration eines Systems zu diskreten Zeitpunkten zum Zwecke der systematischen Steuerung von Konfigurationsänderungen zur Aufrechterhaltung der Vollständigkeit und Verfolgbarkeit der Konfiguration während des gesamten SW-Lebenszyklus.“ [FiBM1998]. Dieser Teilbereich umfasst weiterhin Bereiche wie Änderungsmanagement und Versionsverwaltung.
- *Qualitätsmanagement*: Dieser Teilbereich umfasst Tätigkeiten, die der Sicherstellung der Qualität eines Softwareentwicklungsprozesses dienen. Dabei umfasst die Managementaufgabe die Formulierung von Qualitätsrichtlinien und die Umsetzung in Form operationalisierter Verfahren (z. B. Qualitätshandbücher, Audits).

Die einzelnen Tätigkeitsbereiche werden schließlich durch die zusammenhängende Beschreibung von Aktivitäten und Ergebnissen charakterisiert. Man unterscheidet dabei weiterhin zwischen Aktivität und Aktivitätstyp sowie zwischen Ergebnis und Ergebnistyp

[vgl. FIBM1998]. Diese Unterscheidung entspricht dem Zusammenhang zwischen Vorgehensmodell (als generisches Modell) und Projektmodell (als Ausprägung).

Ein *Aktivitätstyp* lässt sich als abstrakte Beschreibung von Arbeitsschritten oder Tätigkeiten definieren. Eine *Aktivität* ist dann die Ausprägung eines Aktivitätstyps. Analog ist ein *Ergebnistyp* eine abstrahierte Beschreibung eines Resultats des Entwicklungsprozesses und ein *Ergebnis* dessen entsprechende Ausprägung. Ein Ergebnistyp legt zum Beispiel syntaktische oder semantische Qualitätskriterien fest. Beispiele für Ergebnisse sind Prototypen, Berichte oder Produkte.

3.2 Beschreibung und Bewertung von Vorgehensmodellen

Wie bereits in Kapitel 3.1 gezeigt unterscheiden sich Vorgehensstrategien und -modelle sowohl in der Terminologie als auch in Bezug auf die Einsetzbarkeit im Kontext der Erstellung computerunterstützter Lernumgebungen. Nicht jedes Vorgehensmodell kann in beliebigem Anwendungskontext und von jeder Anwendergruppe verwendet werden, da häufig spezifische Domänen festgelegt sind. Um die einzelnen Modelle vergleichen und auf die sinnvolle Einsetzbarkeit schließen zu können, muss zunächst ein Beschreibungsschema identifiziert werden.

Dieser Abschnitt gliedert sich in zwei Teile: *Beschreibungsmerkmale* stellen Vorgehensmodelle in vergleichbarer Form da, während die *Bewertungskriterien* eine Beurteilung der VGM unterstützen. Beide Teile beziehen sich zunächst auf VGM für Softwareentwicklung und werden dann für den Anwendungskontext der Entwicklung computerunterstützter Lernumgebungen analysiert.

3.2.1 Beschreibungsmerkmale

Beschreibungsmerkmale von VGM dienen der übersichtlichen, vergleichenden Darstellung verschiedener Ansätze. Aus diesen Beschreibungsmerkmalen werden Kriterien zur Beurteilung von VGM abgeleitet. Somit ist es möglich, die Einsetzbarkeit von VGM für einen konkreten Anwendungskontext, zum Beispiel für computerunterstützte Lernumgebungen, zu beurteilen.

[Hess1998] unterscheidet vier Dimensionen für die Betrachtung von Vorgehensmodellen:

Die erste Dimension zur Einordnung von VGM ist die *räumliche Dimension*, die in einen Problemraum und einen Lösungsraum unterteilt werden kann [vgl. HeEd1990]. Damit wird eine Abgrenzung hinsichtlich der Tätigkeitsbereiche eines VGM vorgenommen. Es wird dargestellt, in welcher Form der Problemraum beschrieben wird (zum Beispiel in Form einer Anforderungsanalyse) und wie der Lösungsraum beschaffen ist (zum Beispiel in welcher Form die zu entwickelnde Anwendung spezifiziert wird).

Ein weiteres Vergleichskriterium ist die *zeitliche Gliederung*. Diese Dimension erntält den Umfang eines Modells, d. h., welche Tätigkeitsbereiche und Aktivitätstypen erfasst sind und in welchem Zusammenhang diese zueinander stehen (zum Beispiel sequenziell, nebenläufig, parallel).

Als dritte Dimension wird die Dimension der *Organisation* angeführt, die beschreibt, welche Organisationseinheiten an der Softwareentwicklung beteiligt sind (zum Beispiel Manager, Entwickler, Benutzer).

Die letzte Dimension bezieht die *Architektur* ein, die die Verknüpfung von Tätigkeitsbereichen zur Systemarchitektur herstellt. So werden zum Beispiel Verknüpfungen von Phasen zu einzelnen Systemkomponenten oder Softwaremodulen beschrieben.

Die Beschreibungsmethode nach [Hess1998] kann als Basis für eine Beschreibung von VGM dienen. Es ist jedoch anzumerken, dass die oben genannten Dimensionen weitergehend formalisiert werden müssen, um eine Vergleichbarkeit zu gewährleisten.

Eine formalisierte Beschreibungsmethodik wurde in [NoSc1999] vorgelegt. Die nachfolgenden Ausführungen beziehen sich auf die genannte Quelle. Es werden generell sechs Vergleichskategorien herangezogen:

Die *Terminologie* enthält eine Gegenüberstellung der Begriffe, die in den einzelnen Vorgehensmodellen verwendet und auf Begriffe eines Metamodells abgebildet werden (Phase, Aktivität, Ergebnis, Rolle, Technik, Richtlinie/Standard, Notation). Dieses Vorgehen erlaubt einen begriffsunabhängigen Vergleich und fördert zudem die Verständlichkeit eines Vergleiches.

Die *Phasen- und Prozessabdeckung* beschreibt, welche Tätigkeitsbereiche und Aktivitätstypen durch ein Vorgehensmodell abgedeckt werden. Unter Prozessabdeckung versteht man die Vollständigkeit des VGM bezüglich der generischen Prozesse (Tätigkeitsbereiche) Entwicklung, Test, Projektmanagement, Change-/Konfigurationsmanagement und Betrieb/Nutzung. Die Phasenabdeckung beurteilt, welche Aktivitätstypen (Voruntersuchung, Systemanalyse, Entwurf, Erzeugung und Einführung) in einem VGM enthalten sind. Dieses Beschreibungsmerkmal umfasst zudem inhaltspezifische Merkmale. Im Fall der Entwicklung computerunterstützter Lernumgebungen ist die Phasen- und Prozessabdeckung um inhaltspezifische Prozesse zu erweitern.

Der *Beschreibungsumfang* zeigt auf, in welchem Detaillierungsgrad einzelne Aktivitätstypen enthalten sind (Unterteilung in Subaktivitäten oder Techniken).

Die *Prozessarchitektur* bildet die sachlogische und zeitliche Struktur der Tätigkeitsbereiche und Aktivitätstypen ab. Die Prozessarchitektur wird aus der verwendeten Vorgehensstrategie gefolgert. Hinzu kommt die Prozesssteuerung, die beschreibt, ob ein VGM aktivitäts-, ergebnis- oder vertragsorientiert gesteuert und überwacht wird [vgl. Dows1987].

Die *Adaptation* beschreibt, welche Methoden in einem VGM zur Anpassung an einen spezifischen Anwendungskontext gegeben sind. So sind zum Beispiel im V-Modell Anweisungen gegeben, welche Module bei unterschiedlichen Projektgrößen verwendet werden sollen.

Werkzeuge beschreiben schließlich, welche Anwendungen zur Unterstützung, Automatisierung und Durchführung in einem VGM vorhanden sind.

3.2.2 Beschreibungsmatrix für Vorgehensmodelle

Die exemplarisch erläuterten Beschreibungsschemata ermöglichen eine vollständige und umfassende Einordnung von Vorgehensmodellen in verschiedenen Detaillierungsgraden. Auch das Beschreibungsschema nach [NoSc1999] kann offensichtlich noch weiter formalisiert und detailliert werden. So können zum Beispiel für das Merkmal Beschreibungsumfang detaillierte Ausprägungen auf Ebene von Subaktivitäten oder Techniken angegeben werden. Dies ist jedoch nicht sinnvoll, da bei der Analyse verschiedener Vorgehensmodelle ersichtlich ist, dass verschiedene Aktivitätstypen zusammengefasst werden oder implizit enthalten sind. Daher wird der Detaillierungsgrad auf der Ebene des Aktivitätstypen gewählt. Ein Problem sind jedoch die uneinheitlichen Terminologien, die nicht auf den oben erläuterten Ergebnissen von [GeIn2001] beruhen. Vor diesem Hintergrund wird in diesem Abschnitt eine Zusammenfassung der Beschreibungsmerkmale auf Basis dieser Terminologie beschrieben.

Die Merkmale *Vorgehensstrategie* und *Modellierungsparadigma* bieten eine Übersicht über grundsätzliche Charakteristika eines Vorgehensmodells. Diese Merkmale sind zwar in den oben beschriebenen Merkmalen implizit enthalten, bieten jedoch zusätzlich eine Orientierungshilfe zur Einordnung. Die Vorgehensstrategie gibt weiterhin Auskunft über die zeitliche und sachlogische Verknüpfung der Aktivitäten (Prozessarchitektur).

Tätigkeitsbereiche beschreiben die Prozessabdeckung eines Vorgehensmodells. Neben der Systementwicklung im engeren Sinne können die Bereiche Projektmanagement, Konfigurationsmanagement und Qualitätssicherung als komplementäre Komponenten enthalten sein. Gerade im Bereich von Lernumgebungen existieren viele Vorgehensmodelle, die nur Teilbereiche des Entwicklungsprozesses abdecken. Daher muss zur Analyse der Einsetzbarkeit verschiedener VGM detailliert beschrieben werden, welche Bereiche von einem VGM umfasst werden.

Aktivitätstypen beschreiben einzelne Arbeitsschritte des VGM. Dabei werden die originären Aktivitätstypen der analysierten VGM auf die generischen Aktivitätstypen abgebildet: Voruntersuchung, Systemanalyse, Design (Entwurf), Umsetzung (Erzeugung), Test, Implementierung (Einführung) und Nutzung. Soll eine detaillierte Beschreibung erfolgen, so können zusätzlich Subaktivitäten aufgenommen werden.

Ergebnistypen beschreiben zu entwickelnde Produkte, Berichte oder Leistungen, die während des Entwicklungsprozesses vorgelegt werden. Dieses können ebenso Spezifikationen, Prototypen oder Evaluationen sein, die Einfluss auf den weiteren Entwicklungsprozess haben und somit auch als Instrument zur Projektsteuerung dienen.

Rollen beschreiben die Akteure des Entwicklungsprozesses. Einerseits können generische Rollen (Manager, Systemanalytiker, Designer, Fachexperte, Programmierer, Qualitätsbeauftragter, externe Berater), andererseits fachspezifische Rollen in Abhängigkeit vom Kontext definiert werden.

Standards stellen Rahmenbedingungen und Entwicklungsvorgaben durch Spezifikation von Mindestanforderungen oder formalen Regeln dar. Sie spielen gerade in der Systementwicklung und insbesondere bei der zukünftigen Entwicklung computerunterstützter

Lernumgebungen eine wichtige Rolle, um einerseits Qualitätsanforderungen zu erfüllen, andererseits um Faktoren wie Wiederverwendbarkeit oder Interoperabilität sicherzustellen. Daher wird ein Beschreibungsmerkmal eingeführt, dass die verwendeten Standards spezifiziert. Aufgrund der Bedeutung von Standards für computerunterstützte Lernumgebungen werden diese weiterführend in Kapitel 4 erläutert.

Adaptation beschreibt, welche Methoden für eine Anpassung eines Vorgehensmodells zu verschiedenen Projektmodellen vorliegen. Insbesondere bei computerunterstützten Lernumgebungen ist dieser Faktor unerlässlich.

Schließlich werden *Werkzeuge* erfasst, die als Hilfsmittel zur Erfüllung der Aufgaben und Aktivitäten zur Verfügung stehen. Dies können einerseits Hilfsmittel zur Automatisierung sein (z. B. CASE-Tools, Code-Generatoren), andererseits auch Methoden (z. B. Evaluationsmethoden).

Tabelle 12: Beschreibungsmatrix für VGM

| Merkmal/Vorgehensmodell | Beschreibung |
|--|--|
| Vorgehensstrategie | prinzipielle Vorgehensweise eines VGM, zeitliche und sachlogische Verknüpfungen |
| Modellierungsparadigma | Beschreibung der Modellierungsmethode |
| Tätigkeitsbereiche <ul style="list-style-type: none"> • Systementwicklung • Projektmanagement • Konfigurationsmanagement • Qualitätssicherung | Prozessabdeckung eines VGM |
| Aktivitätstypen <ul style="list-style-type: none"> • Voruntersuchung • Systemanalyse • Entwicklung • Erzeugung • Test • Einführung • Nutzung | Aktivitäten des VGM |
| Ergebnistypen | Ergebnisse (Prototypen, Spezifikationen, Berichte), die im Entwicklungsprozess erstellt werden |
| Rollen | beteiligte Aktoren, Rechte und Verantwortlichkeiten |
| Standards | Verwendete Standards und Normen |
| Adaptation | Anpassungsmechanismen zur Erstellung von Projektmodellen |
| Werkzeuge | Verfügbarkeit computerunterstützter Werkzeuge |

3.2.3 Bewertungskriterien für Vorgehensmodelle

Anhand der Beschreibungsmerkmale lassen sich Bewertungskriterien für VGM ableiten, mit deren Hilfe die Einsetzbarkeit in der Entwicklung eines Systems in einem spezifischen Anwendungskontext beurteilt werden kann.

Zu den genannten allgemeinen Kriterien lassen sich aus den in Kapitel 2 beschriebenen Klassifikationskriterien für computerunterstützte Lernumgebungen ebenfalls Beschreibungsmerkmale für spezifische VGM dieses Bereichs identifizieren. Um einerseits die Allgemeingültigkeit der Beschreibungsmatrix nicht zu beschränken, andererseits die spezifischen Besonderheiten bei der Entwicklung computerunterstützter Lernumgebungen zu berücksichtigen, werden die oben beschriebenen Kategorien erweitert und ergänzt.

3.2.3.1 Allgemeine Kriterien der Softwareentwicklung

In der Literatur wird eine Vielzahl von Bewertungskriterien diskutiert [vgl. Balz1982, Raas1993, Hess1998]. Die Priorisierung einzelner Kriterien bei der Bewertung ist jedoch unterschiedlich – in diesem Abschnitt werden einzelne Kriterien erläutert und ihre jeweilige Bedeutung im Anwendungskontext computerunterstützter Lernumgebungen analysiert. [Raas1993] führt neun Kriterien zur Beurteilung von Software auf:

Mit dem Kriterium *Funktionserfüllung* wird beurteilt, ob die entwickelte Software den geplanten Funktionsumfang bietet. Das bedeutet, dass durch Tests vor und während der Implementierung überprüft wird, ob die realisierten Funktionen mit denen der Anforderungsspezifikation übereinstimmen. Die Funktionserfüllung bedeutet jedoch noch nicht, dass die entwickelte Software notwendigerweise eine hohe Qualität aufweist. Eine schlechte Anforderungsspezifikation führt häufig auch zu einer geringen Qualität. Gerade im Bereich von Lernumgebungen ist bei einer fehlenden Einbeziehung der Benutzer während der Anforderungsanalyse der Erfolg einer Lernumgebung nicht sichergestellt.

Die *Effizienz* kann auf Software, Hardware und Netzwerkleistung bezogen werden. Häufig werden dazu Faktoren wie die Ausnutzung von Speicherkapazitäten, Zeitbedarf oder Antwortzeiten verwendet. Während Lernumgebungen nur in Grenzfällen die Kapazitätsgrenzen der aktuellen Rechnergeneration erreichen, ist dieses Kriterium meist nur in Bezug auf die Netzwerkleistung zu betrachten. Gerade bei multimedialen Anwendungen (Videoübertragung, 3D-Grafiken) ist dieses Kriterium vorrangig zu beachten. Eine Betrachtung zur Effizienzüberprüfung findet sich bei [KiCh1990, Kerr2001]. Dabei wird die Effizienz durch monetäre, qualitative und quantitative Maße erfasst. Monetäre Effekte sind zum Beispiel durch eine Kosten-Nutzen-Analyse zu bestimmen. Quantitative Ergebnisse sind durch Testergebnisse, Erfolgsquoten oder Zeitersparnisse messbar, während qualitative Ergebnisse durch Erhebungen über Zufriedenheit oder Image der Bildungsabteilung bzw. Fachgebiete erfasst werden können [Kerr2001].

Die *Zuverlässigkeit* einer Software bedeutet, dass fehlerhafte oder unerwünschte Zustände ausgeschlossen werden sollen. Dieses Kriterium lässt sich durch formale Tests prüfen, wobei eine vollständige Überprüfung des Verhaltens einer Software nicht möglich ist. Die umfangreiche Evaluation muss jedoch Bestandteil jedes Vorgehensmodells sein, da gerade

bei Lernumgebungen fehlerhafte Anwendungen schnell zu einem Motivationsabfall beim Anwender und damit auch zu einem geringeren Lernerfolg führen.

Die *Benutzbarkeit* schließt Kriterien wie Benutzerfreundlichkeit, Ergonomie, Robustheit oder Fehlertoleranz ein. Dieses Kriterium ist innerhalb der Softwareentwicklung von existenzieller Bedeutung. Gerade für Lernende, die eine geringe IT-Vorbildung besitzen, müssen Lernumgebungen einfach und intuitiv benutzbar sein. Die Gestaltung von Benutzerschnittstellen wird daher bei den spezifischen Kriterien für Lernumgebungen aufgegriffen.

Das Kriterium der *Sicherheit* betrifft im Zusammenhang computerunterstützter Lernumgebungen insbesondere Datenschutzaspekte. Einerseits müssen bei der Speicherung der persönlichen Daten der Lernenden die gesetzlichen Richtlinien eingehalten, andererseits müssen Gesetze bezüglich des Urheberrechts und geistigen Eigentums beachtet werden.

Das Kriterium der *Erweiterbarkeit* betrifft die Möglichkeit für Entwickler nach der Fertigstellung einer Software diese um zusätzliche Funktionalitäten zu ergänzen.

Die *Wartbarkeit* umfasst mehrere Kriterien. Der Entwicklungsprozess muss in verständlicher Form dokumentiert sein, um spätere Korrekturen, Änderungen und Erweiterungen vornehmen zu können. Die Wartbarkeit ist ein kritischer Erfolgsfaktor bei der Entwicklung von Lernumgebungen. Gerade im Anwendungsgebiet der Wirtschaftsinformatik verändern sich Lerninhalte innerhalb immer kürzerer Zeiträume. Somit muss eine einfache Möglichkeit zur Verfügung gestellt werden, Lerninhalte, Methoden oder Präsentationsformen zu verändern.

Die Kriterien der *Portabilität und Interoperabilität* beurteilen, ob Software plattformunabhängig genutzt und transferiert werden kann. Einerseits sollen Lernumgebungen ohne zusätzlichen Aufwand auf unterschiedlichen Plattformen verfügbar sein, andererseits sollen bei verteilten Anwendungen die Plattformen nicht auf spezifische Systeme beschränkt sein. Dieser Faktor steht ebenfalls im Zusammenhang mit der Benutzbarkeit, da eine einfache und intuitive Benutzung gerade bei Lernprozessen von besonderer Bedeutung ist: Lernende sollten die Möglichkeit haben, sich auf Lerninhalte zu konzentrieren und wenig Zeit für das Erlernen verschiedener Oberflächen oder Systeme aufwenden.

In engem Zusammenhang damit steht die *Wiederverwendbarkeit*. Die entwickelten Lernumgebungen sollten in einem anderen Kontext ebenfalls nutzbar sein, um die Distributivmöglichkeiten und damit auch die Wirtschaftlichkeit zu verbessern. Insbesondere die hohen Kosten der Entwicklung multimedialer Lerninhalte [vgl. Kerr2001, Schul2001, GIKu2000] machen diesen Aspekt bei der Entwicklung von Lernumgebungen zu einem wichtigen Erfolgsfaktor.

3.2.3.2 Spezifische Kriterien zur Entwicklung von computerunterstützten Lernumgebungen

Die Beschreibungskriterien von Lernumgebungen wurden bereits in Kapitel 2.3 erläutert. Durch die Verwendung dieser Beschreibungsmerkmale kann identifiziert werden, welche Komponenten und Teilbereiche durch ein Vorgehensmodell betrachtet werden. Auch hier findet sich in der Literatur ebenfalls eine Vielzahl von Anforderungen an die Beschreibung

und Beurteilung computerunterstützter Lernumgebungen. Die Bewertung von Vorgehensmodellen zur Erstellung von Lernumgebungen ist wesentlich komplexer als eine reine Beschreibung, da die einzelnen Modelle bezüglich ihrer Zielsetzung, dem Anwendungsfeld und ihrer Methode differieren. In diesem Abschnitt werden Ansätze der *Qualitätssicherung (QS)* und korrespondierende *Qualitätskriterien (QK)* für Lernumgebungen betrachtet und auf Vorgehensmodelle übertragen. Dadurch werden in diesem Abschnitt implizit Anforderungen an ein Vorgehensmodell für computerunterstützte Lernumgebungen postuliert.

Zunächst muss analysiert werden, welche Kriterien in ein Vorgehensmodell einbezogen werden sollten. Dabei müssen folgende Aspekte hinsichtlich der Kriterienauswahl betrachtet werden: Phasenabdeckung, Methodik und Kriterien.

3.2.3.2.1 Phasenabdeckung

Verschiedene Qualitätssicherungsansätze betrachten unterschiedliche Phasen des didaktischen Handelns. Ein umfassender Ansatz zur Qualitätssicherung der Prozesse von Ausbildungsinstitutionen ist zum Beispiel das *European Foundation for Quality Management (EFQM) Excellence Model* [CEHE2001]. Weitere eigenständige Ansätze des umfassenden Qualitätsmanagement für Bildungsinstitutionen sind zum Beispiel das *Quality Mark* der British Association for Open Learning [BAOL2001] oder die Ansätze des Higher Education Funding Council for England [HEFC2001]. Daneben existieren Ansätze, die nur die Qualität von Teilbereichen (z. B. Medieneinsatz in Lernumgebungen) behandeln. Zunächst muss also festgelegt werden, welche Phasen in einem Qualitätssicherungsansatz enthalten sein müssen.

Ein umfassender Ansatz der US Regional Accrediting Commissions [URAC2000] betrachtet Richtlinien für den gesamten Ablauf eines Fernstudiums. Dabei werden fünf Bereiche unterschieden, die sowohl die Administrations- und Entwicklerperspektive als auch die Sicht der Lernenden betreffen:

- *Institutionaler Kontext:* Dieser Bereich stellt Richtlinien und Strategien für eine Bildungsinstitution dar. Darunter fallen zum Beispiel Managementstrukturen und -richtlinien.
- *Curriculum und Methoden:* Dieser Bereich spezifiziert Akkreditierungsrichtlinien zum Design und zur Durchführung von Kursen. Als Beispiel für Methodenanforderungen wird eine angemessene synchrone und asynchrone Interaktion gefordert.
- *Fakultätsunterstützung:* Eine Bildungsinstitution muss einzelne Fakultäten sowohl bei der Kursentwicklung als auch bei internen Geschäftsprozessen unterstützen. So muss zum Beispiel die Ausbildung von Fakultätsmitarbeitern gesichert sein.
- *Studierendenunterstützung:* Dies betrifft Anforderungen über Informationen und Unterstützungsfunktionen, die Studierenden zur Verfügung gestellt werden müssen (z. B. minimale technische Ausstattung, Feedbackzeiten).
- *Evaluation und Prüfungen:* Der Bereich umfasst die Evaluation der Lernleistungen wie auch des gesamten Ausbildungsprogramms. Dabei werden Richtlinien zur Evaluation von Kursen (z. B. Lernerzufriedenheit) und zur Leistungsüberprüfung (z. B. Zwischentests, Abschlussprüfung) beschrieben.

Ein ähnlicher Ansatz wird von [QAAH1999] verfolgt. Hier werden sechs grundsätzliche Richtlinien spezifiziert, die die Bereiche Systemdesign, Programmdesign und -review, Programmübermittlung, Studentenentwicklung und -unterstützung, Studentenkommunikation und -repräsentation sowie Überprüfung unterscheiden. Es werden Richtlinien aufgezeigt, die von Bildungsinstitutionen berücksichtigt werden sollten. Die Richtlinien müssen formalisiert und operationalisiert werden, um in einem Vorgehensmodell verwendet werden zu können.

Dagegen verfolgt zum Beispiel der Ansatz nach [Rekk1998] eine Aufteilung, die sich in vier Bereiche gliedert: Information und Beratung, Kursentwicklung, Kursübermittlung sowie Organisation. Die Bereiche werden weiterhin in vier Phasen (Bedingungen, Implementierung, Ergebnisse und Follow-Up) unterteilt. Diese Klassifikation betrachtet eher Entwicklungs- und Lernprozesse (C-, D-, E-Ebene), während die oben erläuterte Klassifikation übergeordnete Managementstrategien und -richtlinien einbezieht. Diese Ebenen liegen jedoch außerhalb des Wirkungsbereichs von Entwicklungsmodellen. Daher werden im Folgenden die Aktivitäten der C- bis E-Ebene betrachtet.

3.2.3.2.2 Methodik

Grundsätzlich lassen sich Methoden nach verschiedenen Dimensionen unterscheiden. Dabei ist zunächst das Subjekt der *Qualitätssicherung (QS)* zu betrachten. Es können Produkte oder Prozesse evaluiert werden [DEKL2001]. Die *produktorientierte Sicht* entspricht einer Ex-post-Evaluation, d. h., dass das Ergebnis eines Entwicklungsprozesses (z. B. die vollständige Lernumgebung) bezüglich spezifischer Kriterien beurteilt wird. Aus Entwicklerperspektive hat dieses Vorgehen den Nachteil, dass Veränderungen nach Fertigstellung des Produktes aufwendig sind. Das *prozessorientierte Vorgehen* versucht dagegen, die Prozesse, die zur Entwicklung und Produktion eines Produktes führen, zu unterstützen. Als Beispiele dafür seien ISO 9000 und die Anätze aus 3.2.3.2.1 genannt. Im Rahmen dieser Qualitätsnorm werden Prozesse durch Dokumentation und Anforderungen transparent und konsistent gestaltet. Es wird erwartet, dass dadurch entsprechend hochwertige Ergebnisse erzielt werden. Eine Variation ist ein kompetenzbasierter Ansatz. So spezifizieren [CALI1998] Anforderungen an Kompetenzen der Akteure und deren Überprüfung.

Eine weitere methodische Unterscheidung kann nach der Art der Spezifikation getroffen werden. So kann ein QS-Ansatz *Anforderungen* spezifizieren. In diesem Fall werden messbare und nachprüfbar Kriterien definiert, die erfüllt werden müssen. Des Weiteren besteht die Möglichkeit, Prozesse oder Produkte zu dokumentieren und diese *Information* verfügbar zu machen. Dadurch wird eine Transparenz geschaffen und die Entscheidung über die Erfüllung der Qualitätsansprüche wird dem Nutzer bzw. Verbraucher überlassen.

Auch der Einsatz von Vorgehensmodellen kann als QS-Ansatz angesehen werden. Das Vorgehensmodell umfasst verschiedene Richtlinien und soll damit zur Erfüllung spezifischer Kriterien führen. Dabei wird meist ein prozess- und produktorientierter Ansatz kombiniert. Einerseits werden Richtlinien und Unterstützungsfunktionen im VGM zur Verfügung gestellt, andererseits werden die Ergebnisse innerhalb des VGM evaluiert. Durch die Verwendung des VGM stehen ebenfalls prozess- und produktspezifische Informationen zur Verfügung. Der Einsatz von VGM umfasst somit verschiedene methodische Ansätze der Qualitätssicherung.

3.2.3.2.3 Kriterien

Die Operationalisierung von Qualitätssicherungsansätzen erfolgt häufig durch die Bildung von Kriterienkatalogen, die dann in Form von Checklisten (Erfüllung oder Nichterfüllung eines Kriteriums) oder durch die Bildung von Kennzahlen oder Indikatoren vergleichbar gemacht werden. Eine reine ex-post-Überprüfung von Kriterienkatalogen ist dabei nicht sinnvoll, vielmehr sind Kriterien nur Teil eines umfassenden Evaluationsprozesses [vgl. Baum1997, VRee2000, Fric1997] und unterscheiden sich im Detailliertheitsgrad. Während teilweise nur grobe Einteilungen vorgenommen werden, beschreiben andere Modelle detaillierte Anforderungen an Lernumgebungen. Um einen sinnvollen Detailliertheitsgrad bestimmen zu können, werden im Folgenden unterschiedliche Kriterien vorgestellt.

[DEKL2001] identifizieren verschiedene Ebenen von Kriterien aus Sicht von Entwicklern und Lernenden, die die Qualität eines Bildungsprozesses beschreiben (Abbildung 23). Die Verhaltensimplikationen werden in diesem Abschnitt nicht betrachtet.

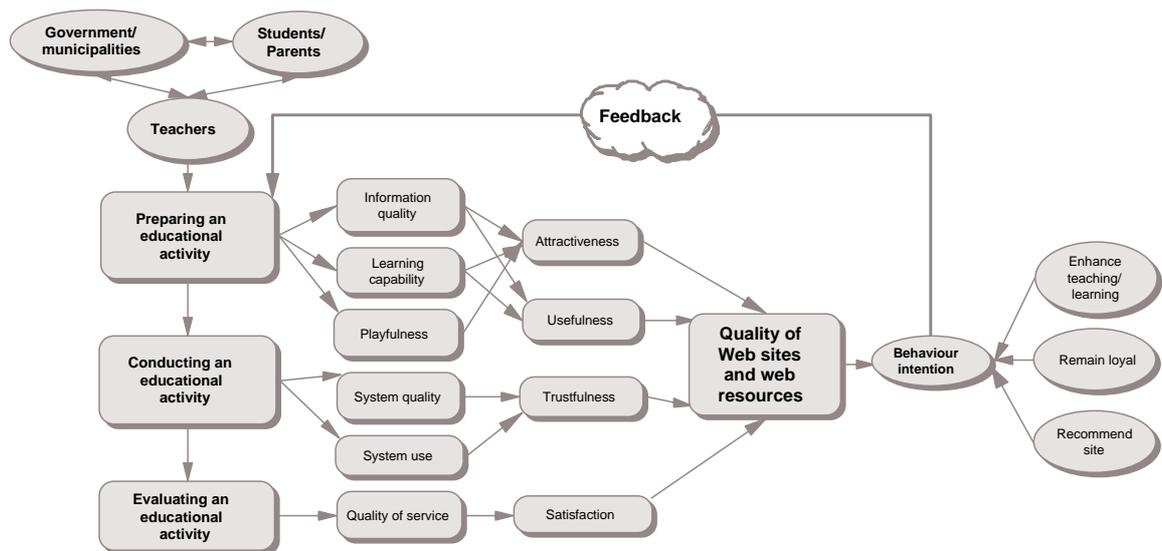


Abbildung 23: Kriterien nach [DEKL2001]

Die Kriterien der zweiten Ebene sind dabei von besonderer Bedeutung: information quality, learning capability, playfulness, system quality, system use und quality of service. Während die letzten drei Kriterien bereits in Abschnitt 3.2.3.1 erläutert wurden, sind die ersten drei spezifisch für Lernumgebungen. Dieses Modell beschreibt zwar grundlegende Kriterien, dennoch wird in [DEKL2001] keine weitere Detaillierung vorgenommen. Der Ansatz kann nur dann sinnvoll genutzt werden, wenn eine Operationalisierung auf Basis der Kriterien erfolgt.

Eine Operationalisierung durch die Festlegung detaillierter Kriterien findet sich bei [ASTD2001]. Der E-Learning Certification Standard umfasst dabei drei Bereiche:

- *Usability* beschreibt die Benutzerfreundlichkeit einer Lernumgebung. Dabei werden folgende Kriterien mit jeweils einem Rating bewertet: Navigation, Orientation, Links, Links Labels, Help, Legibility, Text Production Quality.
- *Technical Standards* beschreiben die technischen Aspekte einer Lernumgebung. Neben den technischen Voraussetzungen werden folgende Kriterien erfasst: Technical Requirements, Install/Uninstall, Reliability, Responsiveness, Exit.
- *Instructional Design* umfasst die didaktische Konzeption einer Lernumgebung. Dabei werden Vorschriften hinsichtlich des Designs vorgegeben: Communicate Purpose, Require Application, Gain Attention and Sustain Interest, Maintain Motivation, Elicit Relevant Knowledge, Show Examples and Demonstrations, Illustrate and Clarify Content, Provide Application Practice, Promote Near-Transfer Learning, Promote Far-Transfer Learning, Provide Integrative Practice Opportunities, Provide Feedback, Near-Transfer Feedback, Far-Transfer Feedback, Offer Instructional Help, Assess Learning, Use Media, Avoid Cognitive Load.

Diese Auflistung zeigt bereits die Schwierigkeit einer detaillierten Aufstellung von Kriterien. Gerade der Bereich des didaktischen Designs ist dabei kritisch zu betrachten. Die angegebenen Kriterien sind teilweise subjektiv und zielgruppenabhängig (z. B. Motivation). Teilweise werden zudem didaktische Vorgaben gegeben, die nicht in allen Lernsituationen sinnvoll sind.

[GIsc1992] fokussieren die Evaluation computerunterstützter Lernumgebungen auf fünf Teilaspekte aus Sicht von Lernendem und Lehrendem: Benutzerakzeptanz, Umfang des erworbenen Wissens, benötigte Studierzeit, Protokoll von Studierverlauf und Wissensdiagnose, Vergleich plausibler Alternativen. Auch diese Betrachtung behandelt nur einen Teilaspekt computerunterstützter Lernumgebungen, kann aber als Anhaltspunkt für diese Perspektiven dienen.

[Kuge2001] beschreibt einen umfassender Kriterienkatalog zur Bewertung aus Entwickler- und Lernendensicht (Tabelle 13):

Tabelle 13: Qualitätskriterien nach [Kuge2001]

| Kategorie | Merkmale |
|---------------------------------|--|
| Handhabung | Installation, Navigation, Programmstabilität, Lerntempo, Bedienbarkeit |
| Darstellung | Aufbau und Gliederung, Darstellung der Inhalte, Ästhetische Gestaltung, Lesbarkeit, Inhalte, Steuerungselemente, Anweisungen, Strukturen der Masken, Schwierigkeitsgrad |
| Didaktischer Aufbau | Aufbaustruktur, Zusatzinformationen, Hilfe-Funktionen, Glossarverknüpfung, Begriffserklärungen, Didaktischer Aufbau, Gliederung der Kapitel/Abschnitte, Texte, Einstiegsniveau, Orientierung am Lernziel |
| Multimedialer Aufbau | Grafiken, Animationen, Farbkombinationen, Grafikerklärungen, Visualisierungen, Realitätsnähe der Animationen, Bildschirmauflösung |
| Audio- und Videosequenzen | Qualität der Audio-/Videosequenzen, Aufbereitung der Audio-/Videosequenzen, Akustik der Sequenzen, Auswahl der Videosequenzen, Bedienung der Audio-/Videowiedergabe, Kombination beider Medien |
| Aufgaben- und Lernzielkontrolle | Fragestellungen, Gestaltung der Aufgaben, Rückmeldungen, Auswertung, Anzahl der Antwortversuche, Überprüfung des Lerninhalts, Markierung richtiger Antworten, Korrigierbarkeit von Eingaben |
| Dokumentation | Qualität der Unterlagen, Handbücher, Begleitmaterial, Gesamterscheinungsbild |
| Einsetzbarkeit | Einsetzbarkeit vor/während/nach einem Kurs |
| Motivation | Individualität, Neugier, ansprechende Wirkung, Nutzen, Einsetzbarkeit, Stress, Akzeptanz, Idee |
| Zusatzfunktionen | Ausdruck, Notizbuchfunktionen, Lesezeichen, Abrufen von Aufgabenlösungen, Korrigieren von Eingaben, Lernzielkontrolle, Bedienung der Audio-/Videowiedergabe, Auswertungen, Abbruch, Deinstallation |

Eine Vielzahl derartiger Kataloge ist sowohl in wissenschaftlicher Literatur als auch bei Bildungsinstitutionen zu finden [vgl. DEKL2001]. Dabei sind die angewendeten Kriterien und die Sichtweisen nicht einheitlich. Bei der Entwicklung eines Vorgehensmodells muss kontextabhängig ein individuelles Qualitätssicherungsverfahren verwendet werden. Damit gewinnt das Kriterium der Anpassbarkeit an Bedeutung. In einem Vorgehensmodell müssen somit die relevanten Bereiche der Entwicklung, Implementierung und Evaluation enthalten sein.

3.2.4 Zusammenfassung der Bewertungskriterien

Im Folgenden werden die zuvor genannten allgemeinen Beschreibungs- und Bewertungskriterien für Vorgehensmodelle der Softwareentwicklung und Vorgehensmodelle für Lernumgebungen zusammengeführt. Das resultierende Bewertungsschema wird anschließend für die Untersuchung einzelner Vorgehensmodelle verwendet. Die Teilbereiche der Vorgehensmodelle für computerunterstützte Lernumgebungen lassen sich wie folgt unterteilen:

- *Management- und Administrationsprozesse* beinhalten die Aufgaben des Managements und der Administratoren in Bildungsinstitutionen. Dabei werden folgende Teilbereiche betrachtet: Termin- und Aufgabenplanung, Kursverwaltung, Monitoring, Tracking, Benutzerverwaltung, Organisation und Kontextanalyse.
- *Entwicklungsprozesse*: Es werden gemäß der Architektur Aktivitäten zur Entwicklung der Inhalts-, Benutzer-, Methoden-, Präsentations-, Kommunikations- und Evaluationskomponente spezifiziert.
- *Lernprozesse* betreffen die Verwendung von Lernumgebungen und beinhalten die Aktivitäten Interaktion, Aktivität, Betreuung, Übermittlung und Prüfung.
- *Evaluations- und Qualitätssicherungsprozesse* umfassen Evaluationsaktivitäten, die nicht durch die Verwendung des Vorgehensmodells impliziert werden (Review- oder Akkreditierungsaktivitäten).

Dadurch ergibt sich eine gegenüber Abschnitt 3.2.2 modifizierte Beschreibungsmatrix:

Tabelle 14: Beschreibungsmatrix für VGM zur Entwicklung von Lernumgebungen

| Merkmal/Vorgehensmodell | Beschreibung |
|--|--|
| Zielsetzung/Zielgruppe | Zielsetzung und Kontext des VGM |
| Vorgehensstrategie | prinzipielle Vorgehensweise eines VGM, zeitliche und sachlogische Verknüpfungen |
| Modellierungsparadigma | Beschreibung der Modellierungsmethode |
| Tätigkeitsbereiche <ul style="list-style-type: none"> • <i>Systementwicklung</i> • <i>Projektmanagement</i> • <i>Administration/Management</i> • <i>Konfigurationsmanagement</i> • <i>Qualitätssicherung</i> | Prozessabdeckung eines VGM |
| Aktivitätstypen <ul style="list-style-type: none"> • <i>Voruntersuchung</i> • <i>Systemanalyse, Kontextanalyse</i> • <i>Entwicklung</i>: Inhalt, Benutzer, Methoden, Präsentation, Kommunikation, Evaluation • <i>Erzeugung</i> • <i>Test</i> • <i>Einführung</i> • <i>Nutzung</i>: Erarbeitung, Interaktion, Betreuung, Übermittlung, Prüfung. • <i>Evaluation</i> | Aktivitäten des VGM, insbesondere Subaktivitäten der Entwicklungs- und Lernprozesse |
| Ergebnistypen <ul style="list-style-type: none"> • Spezifikationen • Prototypen • Lernmanagementsystem • LT-Entwicklungssystem • Lernumgebung | Ergebnisse (Prototypen, Spezifikationen, Systeme), die im Entwicklungsprozess erstellt werden |
| Rollen <ul style="list-style-type: none"> • Lernender • Lehrender • Entwickler • Manager • Domänenexperte • Mediendesigner | am Entwicklungs- und Lernprozess beteiligte Akteure, Rechte und Verantwortlichkeiten |
| Standards <ul style="list-style-type: none"> • Allgemeine Standards • LT-Standards | verwendete Standards und Normen, insbesondere Spezifikation von LT-Standards (siehe Kapitel 4) |
| Adaptation | Anpassungsmechanismen zur Erstellung von Projektmodellen |
| Werkzeuge | Verfügbarkeit computerunterstützter Werkzeuge |

Die einzelnen Aspekte dieser Matrix werden in den Beschreibungen der einzelnen Vorgehensmodelle aufgenommen und diskutiert.

3.3 Vorgehensstrategien und allgemeine Vorgehensmodelle

In diesem Abschnitt werden *Vorgehensstrategien (VGS)* erläutert, die als Grundlage für die Entwicklung von Vorgehensmodellen dienen. Unter Verwendung der Bewertungskriterien aus dem vorangegangenen Abschnitt wird schließlich beurteilt, welche Vorgehensstrategien für die Entwicklung computerunterstützter Lernumgebungen in Betracht gezogen werden können.

Die hier verwendete Klassifikation von Vorgehensstrategien basiert auf dem Kriterium der Prozessarchitektur und daher der Verknüpfung der Aktivitäten. Es ist keine disjunkte Klassifikation möglich, da gerade neuere VGS die Eigenschaften anderer VGS kombinieren. Aufgrund der Verständlichkeit wird auf diese geläufige Klassifikation zurückgegriffen und an entsprechender Stelle auf Überschneidungen verwiesen.

3.3.1 Lineare Vorgehensstrategie

Eine *lineare Vorgehensstrategie* (Abbildung 24) bedeutet, dass die einzelnen Aktivitäten linear miteinander verknüpft sind und sequenziell bearbeitet werden. In jeder Phase wird ein Teilergebnis erzielt (Meilenstein), das nach einer Qualitätskontrolle in die nächste Phase übergeht [Beni1956]. Während beim ursprünglichen *Stagewise Model* nach [Beni56] keine Rückkopplungen vorgesehen waren, wurde das Modell so erweitert, dass bei fehlerhaften Teilergebnissen eine Rückkopplung in die jeweils vorhergehende Phase möglich ist. Rückkopplungen sind aus Komplexitätsgründen auf angrenzende Phasen beschränkt. Bis heute wird die Weiterentwicklung, das *Wasserfallmodell* [Royc1970, Raas1993], häufig verwendet. Auch diese Vorgehensstrategie ist in verschiedene Aktivitätstypen aufgeteilt, die sequenziell bearbeitet werden. Eine Abwandlung des Modells sind Phasen- bzw. Tätigkeitsmodelle nach [Metz1973]. Diese aus dem Bereich des Projektmanagements entstandene VGS erlaubt eine flexiblere Bearbeitung der einzelnen Aktivitäten. So muss eine Aktivität nicht endgültig abgeschlossen sein, um in die nächste Phase überzugehen, sondern es müssen nur Teilergebnisse vorliegen. Die Teilergebnisse (Meilensteine) werden in Form eines Netzwerkes miteinander verknüpft und bieten einen Überwachungsmechanismus zur Projektsteuerung.

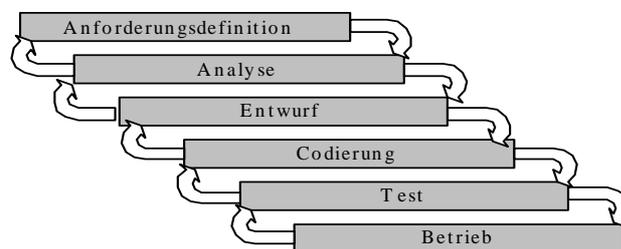


Abbildung 24: Wasserfallmodell

Die beschriebenen Varianten der linearen Vorgehensstrategie ermöglichen eine transparente, verständliche Projektstruktur und sind insbesondere für stark strukturierte, im vorhinein

bekannte Problemstellungen geeignet. Es zeigen sich jedoch gravierende Nachteile [vgl. Balz1996]: Ergeben sich im Verlauf des Entwicklungsprozesses weitere Änderungen, sind diese aufgrund der eingeschränkten Rückkopplungsmechanismen nur schwer zu realisieren, da jeweils nur Rückkopplungen zu vorhergehenden Aktivitäten möglich sind. Weiterhin ist die Einbeziehung der Benutzer nur in der Analysephase vorgesehen, was zu Akzeptanzproblemen bei der späteren Nutzung führen kann. Gerade im Bereich der Lernumgebungen sind diese Faktoren von existenzieller Bedeutung, da motivationale Aspekte insbesondere bei Lernprozessen die Akzeptanz und damit auch das Lernergebnis maßgeblich beeinflussen. Als weiteres Problem ist aufzuführen, dass der sequenzielle Ablauf keine nebenläufigen Aktivitäten zulässt. Die Herstellungszeit wird dadurch häufig verlängert. Auch dieser Faktor ist bei der Entwicklung computerunterstützter Lernumgebungen maßgeblich. Durch die Beteiligung verschiedener Akteure (Entwickler, Mediendesigner, Pädagogen) ist in der Praxis ein hohes Maß an Arbeitsteilung und Kooperation zu beobachten. Es ist offensichtlich, dass in einem solchen Szenario viele Aktivitäten parallel bearbeitet werden müssen, um die Entwicklungszeit gering zu halten.

3.3.2 Iterative Vorgehensstrategie

In einer *iterative Vorgehensstrategie* (Abbildung 25) werden Aktivitäten, ähnlich wie beim Wasserfallmodell, von der allgemeinen Anforderungsspezifikation bis hin zur konkreten Umsetzung als Top-Down-Ansatz bearbeitet [GrSe1995]. Die jeweiligen Ergebnisse gehen nach einer entsprechenden Überprüfung in die nächste Aktivität ein. Bei fehlerhaften Ergebnissen sind Rückkopplungen in jede der vorhergehenden Phasen möglich.

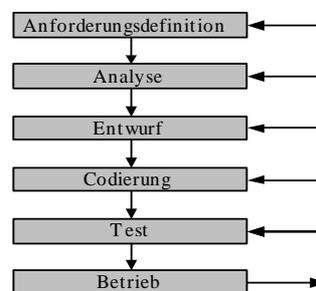


Abbildung 25: Iterative Vorgehensstrategie

Diese VGS erlaubt eine flexiblere Bearbeitung der einzelnen Aktivitäten. Dadurch wird zwar die Überwachung und Steuerung des Entwicklungsprozesses komplexer; dennoch sind auftretende Fehler flexibler zu beheben. Ein solcher Rückkopplungsmechanismus ist unerlässlich, da gerade in der Anforderungsspezifikation häufig Fehler und Missverständnisse entstehen. Dennoch folgt dieses Modell einem starren Ansatz, bei dem Benutzer nur unzureichend eingebunden werden.

3.3.3 Prototyping

Das *Prototypenmodell (Prototyping)* [Balz1998, Ras1993] basiert auf der Verwendung ablauffähiger Modelle und Muster (Prototyp), die der genaueren Definition und Spezifikation von Anforderungen und Benutzerwünschen dienen (Abbildung 26). Die Aktivitäten werden mehrfach durchlaufen, so dass nach der jeweiligen Evaluation des Designs oder der Implementierung die ursprünglichen Anforderungsspezifikationen überarbeitet werden. Das Endprodukt ist in der ursprünglichen Form keine Weiterentwicklung des Prototyps, sondern im Regelfall eine Neuentwicklung.

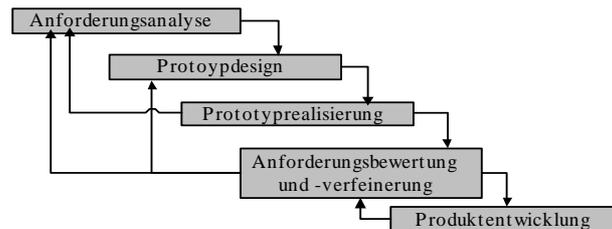


Abbildung 26: Prototyping

Im Vergleich zum Wasserfallmodell ist durch die zusätzliche Entwicklung der Prototypen ein höherer Entwicklungsaufwand zu bewältigen. Weiterhin sind die Koordinations- und Kommunikationsanforderungen an das Entwicklerteam höher. Ein wesentlicher Vorteil dieser VGS ist die frühzeitige Einbeziehung der Endbenutzer. Durch eine laufende Evaluation seitens der Benutzer kann bei diesem Vorgehen die Akzeptanz verbessert werden. Gerade bei Lernprozessen ist die Benutzerintegration dringend notwendig. Dennoch werden im Entwicklungsprozess nur Systeme entworfen, die im weiteren Verlauf nicht weiterverwendet werden und daher zusätzlichen Entwicklungsaufwand bedeuten. Die Weiterentwicklung und -nutzung von Prototypen wird bei der evolutionären VGS betrachtet (siehe 3.3.5).

3.3.4 Nebenläufige Vorgehensstrategie

Nebenläufige Vorgehensstrategien (Abbildung 27) parallelisieren Entwicklungsprozesse durch organisatorische und technische Maßnahmen [Balz1998]. Diese Vorgehensweise findet insbesondere beim Konzept des Concurrent Engineering Anwendung [vgl. Meye1996]. Das zu entwickelnde Gesamtprodukt wird bei der nebenläufigen VGS in einzelne Komponenten unterteilt, die in einem nebenläufigen Prozess parallel entwickelt und implementiert werden. Das Modell wird auch als Clustermodell oder Komponentenmodell bezeichnet [Brem1998].

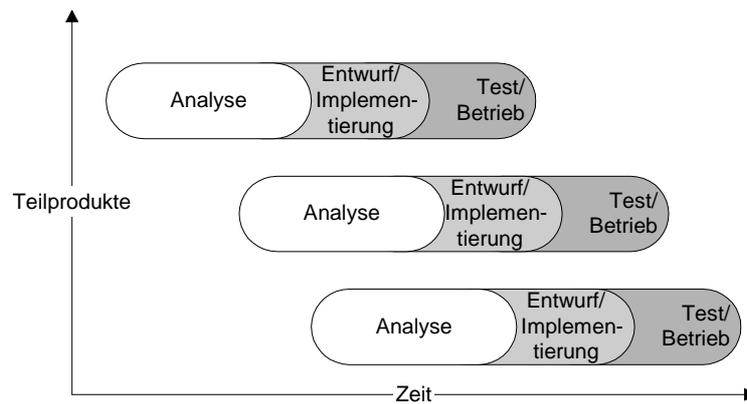


Abbildung 27: Nebenläufige Vorgehensstrategie

Die Vorteile des Modells liegen insbesondere in der leichteren Handhabung der einzelnen Komponenten und in der Zeitersparnis, die durch die Parallelisierung erreicht wird. Risiken entstehen einerseits durch den hohen Koordinationsaufwand, da die einzelnen Komponenten abschließend integriert werden müssen und häufig Fehler erst bei der Fertigstellung des Endproduktes sichtbar werden. Dennoch wird diese VGS bei Lernumgebungen häufig verwendet: Gerade bei der Erstellung von Lernumgebungen, die die Einbindung verschiedenster Inhaltsexperten einschließt, werden die einzelnen Lerneinheiten parallel entwickelt und dann nach der jeweiligen Fertigstellung integriert.

3.3.5 Evolutionäre Vorgehensstrategien

Evolutionäre Vorgehensstrategien (Abbildung 28) orientieren sich am Evolutionsansatz nach [Lehm1980]. Diese Strategie fasst den Entwicklungsprozess als Folge von Entwicklungszyklen auf, die sukzessive zu Endprodukten führen. Dabei endet der Entwicklungsprozess nicht mit der erstmaligen Implementierung der Software; vielmehr ist die Weiterentwicklung (und somit Betrieb und Wartung) integraler Bestandteil dieser Methode. Diese VGS wurde in verschiedenen Weiterentwicklungen aufgegriffen. So folgt zum Beispiel das Capability Maturity Model [Hump1989] ebenfalls einem evolutionären Ansatz. In fünf Phasen (Initial Level, Repeatable Level, Defined Level, Managed Maturity Level, Optimization Level) werden sukzessive verschiedene Reifegrade der Prozess und dementsprechend der Softwareprodukte erreicht. Die Verbesserung der Prozesse, Risikomanagement und kontinuierliche Qualitätssicherung sind integraler Bestandteil dieser Methode.

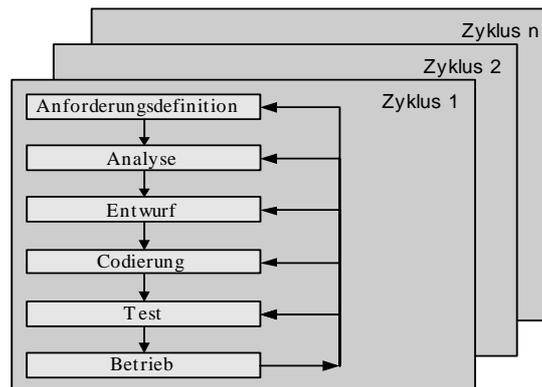


Abbildung 28: Evolutionäre Vorgehensstrategie

Die Vorteile der evolutionären VGS sind insbesondere die frühzeitige Evaluation durch den Anwender. Durch die Weiterverwendung der ersten (prototypischen) Versionen wird der hohe Zeitaufwand der Erstellung von Prototypen verringert. Um jedoch weiterverwendbare Versionen zu erzeugen, ist ein wesentlich höherer Design- und Entwicklungsaufwand notwendig. Ein weiterer Vorteil ist die kontinuierliche Verbesserung während des gesamten Lebenszyklus.

3.3.6 Spiralmodell

Eine Weiterentwicklung bzw. Kombination der oben genannten Ansätze ist das *Spiralmodell*. Durch den wiederholten Durchlauf der einzelnen Phasen unter Einbeziehung von Prototypen werden Änderungen und Rückkopplungen dynamisch einbezogen (Abbildung 29). Zu Beginn jeder Aktivität steht eine Risikoanalyse, in der eine Evaluation des Projektfortschritts durchgeführt und das weitere Vorgehen geplant wird. Das Prozessmodell steht nicht zu Beginn des Projekts fest, vielmehr kann das Vorgehen in jeder Phase flexibel an neue Randbedingungen und Anforderungen angepasst werden.

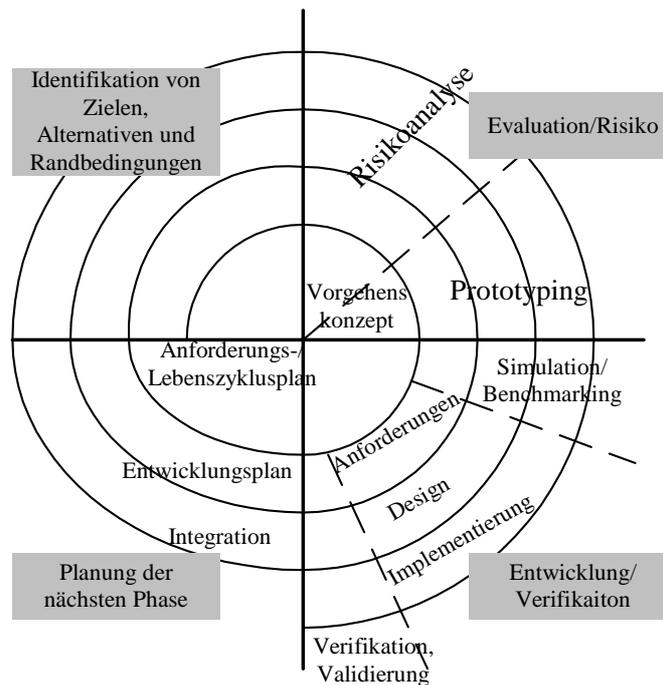


Abbildung 29: Spiralmodell

Der Managementaufwand für die Anpassung des Prozessmodells ist wesentlich höher als bei anderen Vorgehensstrategien. Durch die durchgehende Evaluation und laufende Anpassung werden jedoch Fehler und Risiken frühzeitig erkannt und behoben. Die Flexibilität des Modells rechtfertigt daher insbesondere bei größeren Projekten die Anwendung des Spiralmodells [Balz1998].

3.3.7 Objektorientiertes Modell

Während die bisherigen Vorgehensstrategien nach dem Kriterium der Ablauforganisation klassifiziert wurden, nimmt diese Vorgehensstrategie Bezug auf das Modellierungsparadigma. Zwar kann auch unter Verwendung der bisherigen VGS objektorientiert modelliert werden; aufgrund der derzeitigen Bedeutung objektorientierter Entwicklung wird die *objektorientierte Vorgehensstrategie* jedoch zusätzlich erläutert.

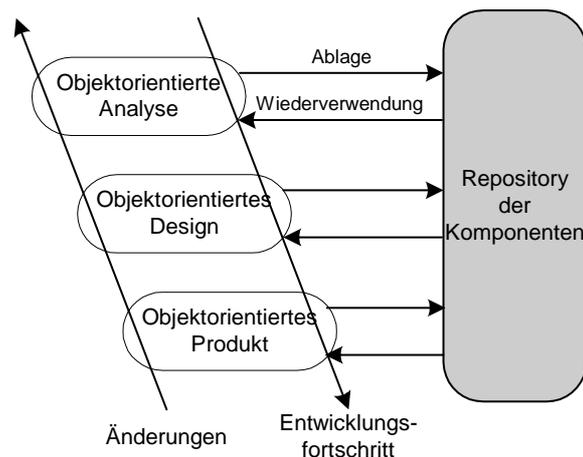


Abbildung 30: Objektorientierte Vorgehensstrategie

Die objektorientierte Vorgehensstrategie besteht aus drei Aktivitäten: Analyse-, Design- und Implementierungsphase. Während der Objektorientierten Analyse (OOA) werden Anforderungen an das Software modelliert. Dies umfasst nach [Balz1996] folgende Teilsysteme:

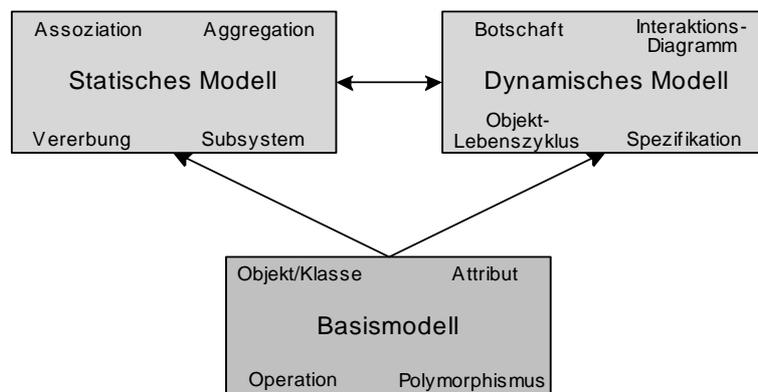


Abbildung 31: Teilsysteme der OOA [Balz1996]

Im Anschluss an die OOA folgt das Objektorientierte Design (OOD), in dem System- und Implementierungsentwurf durchgeführt werden. Dieses umfasst die Festlegung von Klassenbibliotheken und Komponenten sowie der Umsetzung. Entsprechende Rückkopplungen sind zwischen allen Phasen möglich. Die Wiederverwendbarkeit wird durch die Ablage der Analyse- und Designmodelle sowie von Komponenten in einem Repository ermöglicht. Gerade multimedialen Lernumgebungen erfordern häufig aufwendige Medienproduktionen. Um diese wirtschaftlich nutzen zu können, müssen derartige Produktionen wiederverwendet werden. Um diese in anderen Entwicklungsprojekten effizient zu identifizieren

und einbinden zu können, muss ein entsprechendes Repository bestehen, dass die Suche, Retrieval und Verwendung dieser Komponenten ermöglicht.

3.4 Vorgehensmodelle zur Entwicklung computerunterstützter Lernumgebungen

Für den Anwendungskontext der Entwicklung computerunterstützter Lernumgebungen wurde eine Vielzahl von Vorgehensmodellen unterschiedlicher Fokussierung entwickelt.

- *Didaktische Modelle* setzen einen Schwerpunkt auf didaktischen Methoden. Bereits in Abschnitt 2.2 wurde das Berliner Modell gezeigt, das ebenfalls als Vorgehensmodell verwendet werden kann. Didaktische Modelle bestimmen den Technologieeinsatz als Teil einer Methode [vgl. Moon2001]. Dieses wird am Beispiel von Vorgehensmodellen des Instruktionsdesign erläutert (siehe 3.4.1).
- *Technologieorientierte Modelle* folgen der Annahme, dass die Technologie bzw. Medieneinsatz zentraler Untersuchungsgegenstand ist und den Lernprozess maßgeblich beeinflusst. In diesem Abschnitt werden exemplarisch Hypermedia-, CBT- und ein generisches technologieorientiertes Modell vorgestellt, um die Charakteristika aufzuzeigen.
- Die Trennung didaktischer und technologieorientierten Ansätze ist nicht immer strikt vorzunehmen. So ist eine Vielzahl *hybrider Modelle* entstanden, auf die abschließend eingegangen wird.

3.4.1 Instruktionsdesign

Eine wichtige Klasse didaktischer Vorgehensmodelle orientiert sich am *Instruktionsdesign (ID)*. Unter Instruktionsdesign versteht man die systematische Entwicklung von Instruktionsmaterial im engeren und Instruktionsprozessen im weiteren Sinne [vgl. Issi1997, Blum1998]. Das Instruktionsdesign geht auf [Gagn1965] zurück und integrierte behavioristische und kognitivistische Lerntheorien. Diese grundlegende Arbeit führte zur Entwicklung der Instruktionstheorie [GaBW1979] und ist die Grundlage für die Ausgestaltung und Organisation von Lernprozessen zur Erreichung bestimmter Lernziele. [ReBM1994] fassen die zentralen Einflussfaktoren des Instruktionsdesigns zusammen:

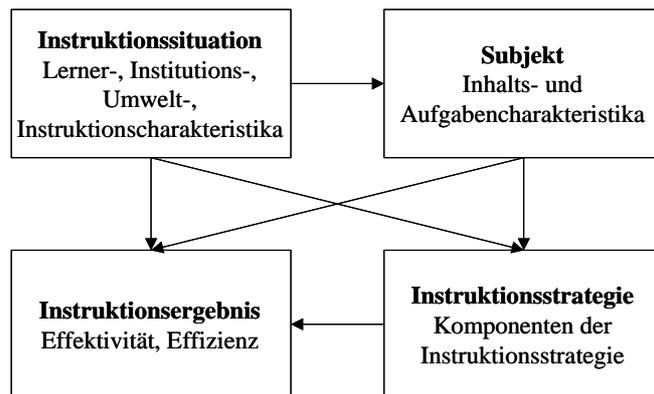


Abbildung 32: Einflussfaktoren des Instruktionsdesigns

Diese Einflussfaktoren stimmen größtenteils mit den in Abschnitt 2.3 diskutierten Faktoren überein. Die Vorgehensweise des Instruktionsdesigns unterscheidet sich jedoch von anderen Theorien und Modellen. Aufgrund der *Instruktionssituation* werden Parameter bezüglich des *Subjekts* vorhergesagt und angepasst. Daraufhin wird eine *Instruktionsstrategie* festgelegt, die wiederum auf das Ergebnis schließen lässt. [ReBM1994] verfolgen damit ein präskriptive Strategie, bei der zum Beispiel die Variablen der Instruktionssituation als unabhängige Variablen angesehen werden, die durch Anwendung eines systematischen Regelsystems zu optimalen Instruktions-, Präsentations- und Managementstrategien und entsprechenden *Instruktionsergebnissen* führen. [Issi1997] fasst verschiedene Modelle des Instruktionsdesign zusammen:

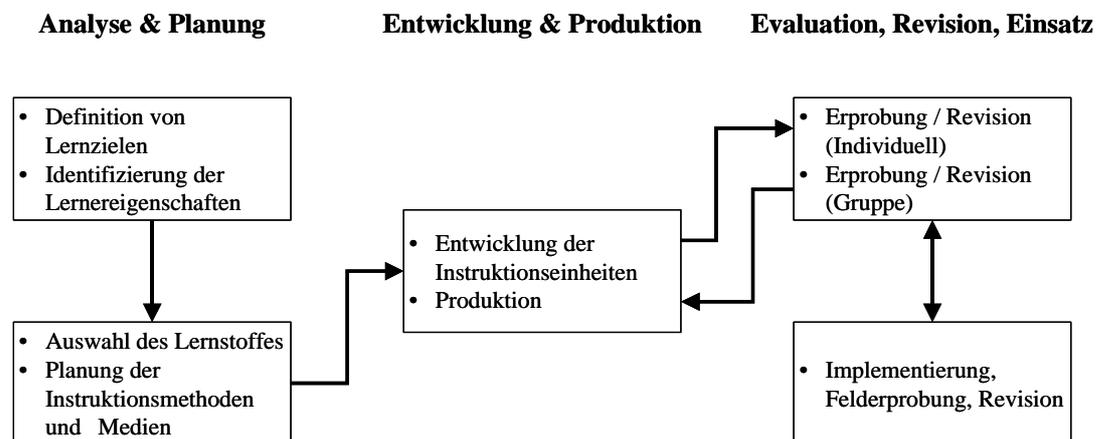


Abbildung 33: Phasen des ID nach [Issi1997]

Das Instruktionsdesign bezieht sich einerseits auf Theorien des Lernens und Lehrens, andererseits auf Erfahrungen, die zu Modellen formalisiert werden. Als wichtige Fragestellung diskutieren [SeDi1997], unter welchen Bedingungen Entwickler und Lehrende valide Theorien anstelle der naiven Konzepte einsetzen. Es wird weiterhin hervorgehoben, dass die Planungsaufgaben bezüglich der Analyse, der Abfolge der Instruktionsprozesse und der

Methoden zu einer komplexen Aufgabe werden, die durch die Technologieunterstützung und damit implizit durch Vorgehensmodelle unterstützt werden muss.

Als Beispiel für ein Modell des Instruktionsdesigns sei die *Component Display Theory (CDT)* nach [erstmalig in Merr1983, zitiert nach Merr1994] genannt. Ziel der CDT ist einerseits die Abbildung und Beschreibung von Lernergebnissen und Inhalten (descriptive CDT), andererseits die Beschreibung des Prozesses, wie diese Ergebnisse erzielt und überprüft werden können (prescriptive CDT).

Ausgangspunkt der CDT ist die *Performance-Content Matrix (P/C-Matrix)*. In dieser Matrix werden Lernergebnisse in die Kategorien *Find*, *Use*, *Remember Generality* sowie *Remember Instance* unterteilt. Die Lerninhalte werden durch die Kategorien *Fact*, *Concept*, *Procedure* sowie *Principle* beschrieben. Ein exemplarischer Ausschnitt wird in Tabelle 15 gezeigt: Ein Lerninhalt besteht darin, ein Konzept anwenden zu können (Use Concept). In der P/C-Matrix wird nun das Vorgehen (Condition) festgelegt, das aus der Demonstration eines Ereignisses besteht. Nach dieser Demonstration soll der Student eine Klassifikation durch Sortierung vorgegebener Attribute durchführen (Behavior) und das Verhalten durch eine geringe Fehlerquote belegen (Criterion). Neben diesem allgemeinen Vorgehen werden das optimale Präsentationsformat (Presentation) und geeignete Übungen (Practice) vorgegeben.

Tabelle 15: Performance-Content Matrix nach [Merr1994]

| | Fact | Concept | Procedure | Principle |
|----------------------------|-------------|--|--|------------------|
| Find | | | Condition: Demonstration eines Ereignisses Behavior: Ableitung der Einzelschritte durch Analyse des Experiments Criterion: Demonstration | |
| Use | | Condition: Zeichnung / Abbildung eines neuen Beispiels Behavior: Student kann Klassifikation durch Sortierung durchführen Criterion: Wenige Fehler | | |
| Remember Generality | | | | |
| Remember Instance | | | | |

Die CDT ist somit ein Vorgehensmodell, das sich auf die Entwicklung von Lernumgebungen und deren Qualitätssicherung spezialisiert. Ein Schwerpunkt wird auf die Systemanalyse (Inhalts-, Lerner-, und Kontextanalyse) gelegt, entsprechende Lehrmethoden und Prä-

sentationsformate werden abgeleitet. Das Modell ist stark phasenorientiert, während in Weiterentwicklungen komplexere Rückkopplungen zugelassen werden. Ergebnis des VGM ist eine genaue Spezifikation des Lehrprozesses. Weiterhin steht das Tool *ID Expert* zur Verfügung, um Lernumgebungen nach den Prinzipien der CDT zu implementieren. Eine genauere Beschreibung des ID Expert findet sich bei [Merr1987, Merr1993, LiMe1994a, LiMe1994b]. Eine Erweiterung und Veränderung dieses Modells ist das Instructional Design der zweiten Generation (ID₂) [Merr1990], in der die Transaktionstheorie eingeführt wird. Unter einer Transaktion wird eine instruktionale Interaktion mit Lernenden verstanden, die verschiedene Kommunikationsformen, Präsentationsmethoden und Inhalte einschließt. Während zuvor Präsentationsformen im Vordergrund standen, werden nun Interaktionen und die Kommunikation als Integration konstruktivistischer Methoden bezeichnet. [Schu1996] kritisiert jedoch, dass diese Methoden verfälscht werden und weiterhin eine präskriptive Vorgehensweise bestehen bleibt. [Weid1993] dagegen bezeichnet das Vorgehen als pragmatische Zwischenposition.

[CeAC1996] erweitern den Entwicklungsprozess um konstruktivistische Aspekte, indem soziale Aspekte, Reflexivität und Lernorientierung im Designprozess sowie vor allem multiple Perspektiven im Lernprozess einbezogen werden. [Will2000] betont im *Recursive and Reflective Design and Development Model*, dass Designprozesse rekursiv, iterativ und partizipativ sein müssen.

Der Erfolg der ID-Modelle hängt von verschiedenen Faktoren ab. Es ist in Frage zu stellen, ob Faktoren wie Lernmotivation, Lernziele oder Lernpräferenzen objektivierbar, kontrollierbar und vorhersagbar sind. Die Sicht des ID widerspricht daher den wesentlichen Ergebnissen des Konstruktivismus. [Schu1996] kritisiert zudem, dass Instruktion weiterhin als lehrerzentriert und -kontrolliert angesehen wird. Ein weiterer Kritikpunkt ist der Zeitaufwand, der für die Modellierung und Formalisierung in Anspruch genommen wird. [Blum1998] weist ferner auf den Zeitaufwand für ungeplante Prozesse hin. Besonders beim wichtigen Kriterium der Wiederverwendbarkeit ist eine Formalisierung und Operationalisierung unerlässlich.

Die Kritik richtet sich damit nicht gegen den Versuch der Abbildung von Einflussfaktoren, sondern vielmehr gegen das präskriptive, regelbasierte und insbesondere kontrollierbare System, das das Instruktionsdesign vorgibt. Die Methoden und Erfahrungen des Instruktionsdesigns können als Leitfaden dienen, da in diesen Theorien Erfahrungen für bestimmte Lernsituationen abgebildet werden.

Für die Entwicklung des Essener-Lern-Modells werden aufgrund dieser Kritik und aufbauend auf die Schlüsse aus 2.3.4.4 zwar kognitivistische bzw. konstruktivistische Konzepte empfohlen. Es ist jedoch die primäre Aufgabe, einen Ansatz zu entwickeln, der Rahmenbedingungen abbildbar macht und Akteure Hilfsmittel zur Verfügung stellt. So können sowohl konstruktivistische als auch Methoden des Instruktionsdesigns abgebildet werden. Es wird also nicht eine Lerntheorie oder Methode vorgeschrieben. Vielmehr werden die Möglichkeiten von Entwicklern und Designern erweitert, unabhängig welche Lerntheorie oder -methode verwendet bzw. bevorzugt wird.

3.4.2 Technologieorientierte Vorgehensmodelle für Lernumgebungen

Im Folgenden werden technologieorientierte VGM beschrieben. Zunächst werden Modelle beschrieben, die spezifische Komponenten von Lernumgebungen betrachten und nur partiell einsetzbar sind. Daraufhin folgt die Beschreibung eines umfassenden Modells.

3.4.2.1 CBT-Vorgehensmodell

Das CBT-Vorgehensmodell nach [Step1989] wurde ursprünglich zur Entwicklung multimedialer CBT-Anwendungen konzipiert und wird auch heute häufig für die Entwicklung computerunterstützter Lernumgebungen verwendet. Das VGM gehört zur Klasse drehbuchorientierter Vorgehensmodelle und stellt exemplarisch Vorgehensmodelle dar, die nur für spezifische Lernumgebungstypen eingesetzt werden können. Das sequenzielle Modell besteht aus vier Aktivitäten: Lernprogrammplanung, -erstellung, -erprobung und -distribution. Die Lernprogrammplanung umfasst jedoch weniger die spezifische Planung der Lernumgebung oder Curricula, sondern den Tätigkeitsbereich der Projektplanung. Bei der Lernprogrammerstellung werden Lernziele und daraufhin Lerneinheiten geplant. Der Ablauf der Lerneinheiten wird in einem Drehbuch spezifiziert, das als Vorlage für die Erstellung durch Programmierer und Designer dient. In der Erprobungsphase werden zunächst die technische Verifikation und Validierung vorgenommen, dann wird ein Feldtest mit einer Testgruppe durchgeführt. Schließlich umfasst das VGM noch die Distribution der Lernprogramme (Abbildung 34).

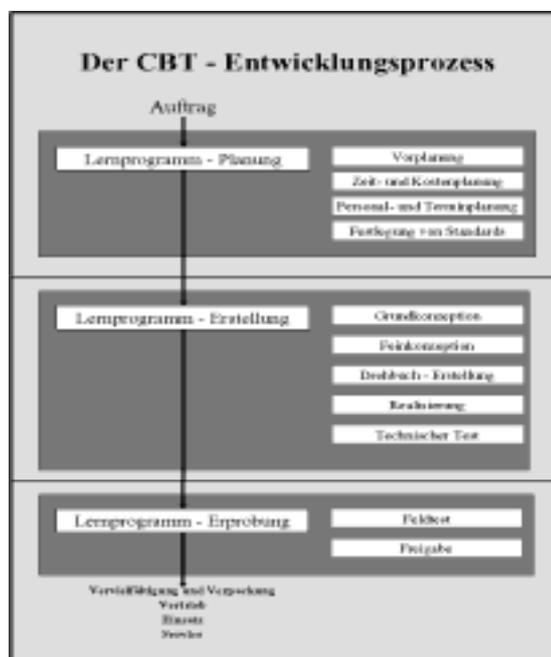


Abbildung 34: CBT-Vorgehensmodell [Step1989]

Das CBT-VGM ist aufgrund einer klaren Gliederung der Aktivitäten und Spezifikationen für die Erstellung von einfachen Lernumgebungen sinnvoll einsetzbar. Die Kooperation zwischen den Akteuren des Entwicklungsprozesses ist klar strukturiert. Dennoch eignet sich das Modell eher für klassische Drill & Practice-Anwendungen, die drehbuchorientiert ablaufen. Lernumgebungen, die mit diesem VGM entwickelt werden, beziehen den Lernenden weder in die Planung noch in die Entwicklung in ausreichendem Maße ein. Die Umsetzung komplexer konstruktivistischer Lernumgebungen ist mithilfe dieses Modells nicht möglich. Zur Realisierung einzelner Sequenzen innerhalb einer Lernumgebung (z. B. Videosequenzen) ist das Modell dennoch effizient nutzbar.

3.4.2.2 Hypermedia-Vorgehensmodelle

Eine Vielzahl von VGM befasst sich mit Richtlinien und Modellierungsmethoden für spezifische Anwendungen und Teilbereiche des Designs, der Entwicklung und der Implementierung von Lernumgebungen. Wie bereits erläutert, werden Hypertextanwendungen häufig mit Lernumgebungen gleichgesetzt. Modelle zur Entwicklung von Hypertext- und Hypermediaanwendungen werden häufig im Kontext der Lernumgebungsentwicklung angewendet. Sie können zwar nicht zum Entwurf einer kompletten Lernumgebung verwendet werden, sind aber gerade für die Repräsentations- und Präsentationssicht sinnvoll einsetzbar. In diesem Abschnitt werden, in Anlehnung an [BBBD1996], drei Modelle zur Entwicklung von Hypermediaanwendungen vorgestellt, um Potenziale und Defizite dieser Klasse darzustellen. Generell lassen sich bei der Hypermedia-Entwicklung drei Ebenen unterscheiden [BBBD1996]:

- Der *Entwurf des logischen Grundgerüsts* beinhaltet die Modellierung von Schemata, Instanzen und Leseschemata.
- Der *Entwurf der Nutzung* spezifiziert konkrete Zugriffsstrukturen, Inhalte, Navigationsunterstützung und Zusätze einer Anwendung.
- Der *Entwurf der Präsentation* bestimmt schließlich das Layout einer Hypertextanwendung.

3.4.2.2.1 Hypertext Design Model (HDM)

Das *Hypertext Design Model (HDM)* ist ein Modell zur Strukturierung von Hypertextanwendungen [GaPS1993] und dient insbesondere zum Entwurf des logischen Grundgerüsts. Ohne Berücksichtigung konkreter Inhalte werden globale Klassen, Informationsobjekte und eine Navigationsstruktur entworfen. Die Struktur wird wie folgt beschrieben:

Zunächst werden in *Objektklassen* ähnliche Objekte zusammengefasst und modelliert. Objekte beschreiben Informationsstrukturen und werden als kleinste autonome Einheiten einer Anwendung definiert. Objekte bestehen aus Komponenten, die wiederum in Einheiten unterteilt sind. In dieser Form werden Informationsstrukturen beschrieben, hinzu kommen die *Beziehungen der Objekte* in Form von Verweisen: Perspektivische Verweise stellen Beziehungen zwischen den Einheiten einer Komponente her; strukturelle Verweise beschreiben Beziehungen zwischen Komponenten innerhalb eines Objektes; Anwendungs-

verweise stellen Beziehungen zwischen Objekten und deren Komponenten her. Strukturelle und perspektivische Verweise werden größtenteils aus der Objektstruktur abgeleitet.

Durch die Definition der beschriebenen Strukturen wird ein HDM-Schema abgebildet, das die Menge zulässiger Objektklassen und Verweisklassen beschreibt und entsprechende Bedingungen für Instanzierungen enthält. Somit erfolgt bei der konkreten Anwendungsentwicklung eine partielle semantische Überprüfung der Inhalte. Zugriffsstrukturen stellen ein weiteres Hilfsmittel zur Anwendungsentwicklung dar; sie können als Eintrittspunkte in die Hypertextstruktur angesehen werden (wie etwa Guided Tours). Die Erstellung einer Anwendung erfolgt durch die Instanzierung der Objekte und Verweise sowie durch die Erstellung einer Browsing Semantic. Durch die Anwendung des HDM-Schemas können dann verschiedene Anwendungen aus bestehenden Strukturen entwickelt werden. Die Weiterentwicklung HDM2 schlägt die Einführung von Navigation Patterns vor [GaPS1993], die Methodologie wird jedoch nicht weiterreichend verändert.

Durch die Verwendung des HDM wird verschiedenen Akteuren eine gemeinsam verwendbare Modellierungsmethode vorgegeben, die zur exakten Definition einer Anwendung führt. Durch die Verwendung der abstrakten Objekt- und Verweisklassen werden Strukturen zudem für weitere Anwendungsentwicklungen wiederverwendbar gemacht.

3.4.2.2.2 Relationship Management Methodology (RMM)

Während das HDM das Vorgehen nur implizit modelliert, ist die *Relationship Management Methodology (RMM)* ein VGM zur Entwicklung von Hypermediaanwendungen, das auf die klassischen Phasen der Softwareentwicklung zurückgreift:

- Während des *Entity-Relationship (ER)-Designs* werden die grundlegenden Entitäten, Attribute und Beziehungen des Anwendungsgebiets in Form von ER-Modellen spezifiziert.
- Während des *Entity-Designs* werden Informationen und Sichten auf die Informationen festgelegt, die präsentiert werden sollen. Hinzu kommt die Festlegung von Assoziationen durch uni- und bidirektionale Verweise. Dies entspricht den strukturellen Verweisen des HDM.
- Beim *Navigations-Design* werden assoziierten Verweisen Navigationsprimitive (Uni- und bidirektionale Links, Index, Guided Tour, indizierte Guided Tour, Auswahl) zugeordnet.
- *Konvertierungsprotokoll-Design*: In dieser Phase werden die Primitive regelbasiert in eine konkrete Implementierung transferiert (z. B. HTML).
- Beim *Benutzerschnittstellen-Design* wird das Layout der Präsentation festgelegt.
- Das *Laufzeitverhalten-Design* bestimmt dynamische Aspekte der Präsentation.
- In der *Implementierungsphase* erfolgt die Umsetzung der bisher festgelegten Spezifikationen.

Die RMM betrachtet somit insbesondere die Design- und Implementierungsphase, während Systemanalyse und Evaluation nur erwähnt werden und das Modell somit in ein Gesamtmodell integriert werden muss. Die Verfügbarkeit von Primitiven ermöglicht eine vereinfachte Konzeption von Hypertextanwendungen, allerdings sind weitere Schwächen zu erkennen [BBBD1996]. So werden nur begrenzte Primitive zur Verfügung gestellt; weiterhin ist keine kontextabhängige Navigation möglich.

3.4.2.2.3 Modell zur Entwicklung hypermedialer Lernumgebungen

Ein weiteres Modell, das die oben aufgeführten Schwächen der Hypermedia-VGM überwinden soll, ist das *Modell zur Entwicklung hypermedialer Lernumgebungen* nach [Blum1998]. Die vier Hauptphasen des iterativen Entwicklungsprozesses sind die Bedarfsanalyse, Alternativenentwicklung, Produktion und Anwendung/Evaluation. Neben diesen Phasen sind durchgehend Projektmanagement- und Revisionsprozesse durchzuführen (Abbildung 35). Fokus dieses Modells ist die Produktion von hypermedialen Lernumgebungen. Zusätzlich ist ein Prototypingzyklus vorgesehen.

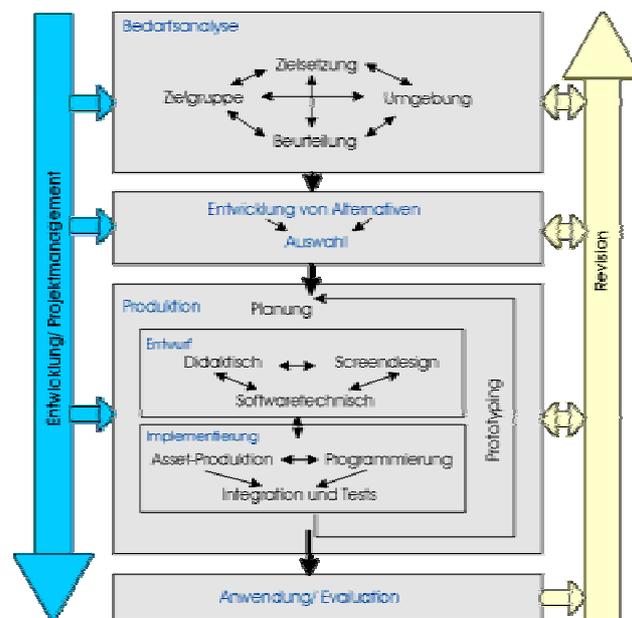


Abbildung 35: Entwicklungsmodell für hypermediale Lernsysteme [Blum1998]

Das Modell beinhaltet die wichtigsten Elemente der didaktischen und technologischen Planung (siehe Abschnitt 2.2) von Lernumgebungen und umfasst die D- und E-Ebenen der didaktischen Planung. Dennoch ist festzustellen, dass die umfassende Planung eines Einsatzes computerunterstützter Lernumgebungen nicht möglich ist, da die Planung und Entwicklung von Curricula nicht vorgesehen ist. [Blum1998] erwähnt dagegen die curriculare Einbindung, um Lernumgebungen in einen organisatorischen Kontext einzubetten. Vernachlässigt wird allerdings, dass der Einsatz von Lernumgebungen häufig zu Änderungen oder kompletten Neuplanungen des Curriculums führt und somit auch diese Phase unterstützt werden muss. Im Gegensatz zu anderen Modellen ist die Evaluation nicht nur als

abschließende Evaluation enthalten, sondern es werden bereits in den ersten Planungsphasen Evaluationsinstrumente festgelegt.

Zur Gestaltung von Lernumgebungen und zur Förderung der Wiederverwendbarkeit ist der Einsatz von Hypertext- und Hypermedia-Vorgehensmodellen durchaus gerechtfertigt. Da jedoch diese Klasse von VGM nur einen Teilausschnitt der Entwicklung von Lernumgebungen betrachtet, ist die Integration in ein umfassendes Modell unerlässlich. Der Ansatz von [Blum1998] ist ein erster Schritt, die didaktische und technologische Planung gleichberechtigt zu betrachten und weitere Phasen abzudecken. Dennoch verlangt auch dieses Modell eine Erweiterung, um eine umfassende Planung von Lernumgebungen vornehmen zu können.

3.4.3 Hybride Ansätze für Lernumgebungen

Hybride Ansätze integrieren aktuelle pädagogische wie technologische Aspekte und sollen so die Schwächen der erläuterten VGM beheben. In diesem Abschnitt werden Ansätze vorgestellt, die den gesamten Entwicklungsprozess aus didaktischer und technologischer Perspektive unterstützen.

3.4.3.1 Hybride Vorgehensmodelle

Die Klasse hybrider Modelle zur Entwicklung von Lernumgebungen beinhaltet das *Vorgehensmodell zur Erstellung virtueller Bildungsinhalte* nach [KlSt2001]. Dieses Modell stellt den Aspekt der Entwicklung wiederverwendbarer Komponenten in den Vordergrund und orientiert sich am Wasserfallmodell. Folgende Aktivitäten werden betrachtet:

- In der *Analysephase* werden Lernziele, Lernereigenschaften und Lerninhalte in natürlicher Sprache modelliert.
- In der *Designphase* werden Lerninhalte durch Module und entsprechende Beziehungen modelliert, die Entwürfe von Lehrstoff, Lernszenarien, Benutzerschnittstellen und Datenmodellen umfassen.
- Während der *Implementierung* erfolgt die Implementierung der Kursstruktur, Navigation und Multimedia-Einheiten. Hinzu kommt die Spezifikation von Metadaten zur Beschreibung der Lerninhalte (siehe auch 4.4.3.1).
- *Einsatz/Wartung*: Die Evaluation erfolgt in diesem Modell im Anschluss an die Implementierungsphase und wird durch Befragungen und Zugriffsstatistiken durchgeführt. Daraufhin erfolgen entsprechende Maßnahmen zur Wartung.

Das Modell ermöglicht zwar die strukturierte Entwicklung wiederverwendbarer Lernmodule, dennoch ist schon die Vorgehensstrategie kritisch zu betrachten. Durch die sequenzielle Vorgehensweise werden Lernende nicht in den Entwicklungsprozess einbezogen. Ebenso ist der Qualitätssicherungsprozess nur als Ex-post-Evaluation enthalten. Weiterhin

wird angemerkt, dass keine Analyse- und Designwerkzeuge zur Verfügung gestellt werden. Somit ist die Wiederverwendbarkeit auf die Implementierungsebene beschränkt.

Ein flexibler Ansatz ist das VGM nach [Seuf1999, SeBH2001]. Das *Plato-Cookbook* zielt durch die Form der Präsentation auf einen weiteren wichtigen Aspekt ab: die Benutzbarkeit durch Entwickler und Lehrende. Während technologieorientierte Ansätze bei pädagogisch orientierten Benutzern wenig Akzeptanz finden, bestehen bei Entwicklern häufig Defizite in der didaktischen Ausbildung. Daher wird in diesem Ansatz die Metapher des Kochbuchs aufgegriffen, um die Inhalte verständlich zu gestalten. Das VGM umfasst einen verständlichen Leitfaden für die Kurserstellung (C- und D-Ebene), der für Anfänger unterschiedlicher Domänen konzipiert ist. Auch wenn in Frage gestellt werden kann, ob wenig ausgebildete Aktoren in der Lage sind, hochwertige Lernumgebungen zu erstellen, so ist dieses Modell umso mehr gerechtfertigt, um intuitive Trial-and-Error-Entwicklungen zu vermeiden. Die Meinung teilt auch [Eule1992]: „Entwicklungsmodelle können immer nur Wegweiser sein, sie können nicht die Bewältigung des Weges selbst leisten.“ Dieser Aspekt muss beachtet werden, da Partizipation, Flexibilität und Kreativität wichtige Akzeptanzkriterien für Entwickler und Lehrende sind. VGM sollten daher nicht einschränken, sondern neue Möglichkeiten für die Aktoren bieten.

3.4.3.2 3-Space-Design-Strategy

Die *3-Space-Design-Strategy* nach [Moon1999, Moon2000, Moon2001] versucht Aspekte des Konstruktivismus, der Benutzerpartizipation und der prototypischen Entwicklung zu integrieren. Dabei werden drei Design-Räume (Design-Spaces) unterschieden. Die Terminologie impliziert, dass jeder der drei Design-Räume gleichberechtigt ist und keine sequenzielle Abfolge besteht. Ziel ist es, eine Strategie zu entwickeln, die strukturiertes mit assoziativem Vorgehen verbindet [Moon2001]. Strukturierte Vorgehensmodelle bauen auf das Prinzip der Dekomposition auf (z. B. Lineares Modell, Wasserfallmodell), während assoziative Vorgehensmodelle die Beziehung zwischen Produkt und Nutzer in den Vordergrund stellen (z. B. Evolutionäre Modelle, Prototyping). Folgende Prinzipien wurden bei der Entwicklung berücksichtigt:

- Benutzerkontext steht bei der Entwicklung von Lernumgebungen im Vordergrund
- Beachtung der internen wie externen Akzeptanz
- Aushandlung temporärer Konsenszustände
- Prototyping zur Kontextualisierung und Reduktion von Unsicherheitsfaktoren
- Entwicklung von Teilprodukten zur Anpassung durch die Benutzer

Durch diese Prinzipien werden insbesondere die Lernenden und deren Kontext in den Entwicklungsprozess einbezogen, ohne dabei die Entwicklerperspektive zu vernachlässigen. Die Spezifikation von Konsenszuständen und -spezifikationen ermöglicht zudem ein strukturiertes Vorgehen. Den Anforderungen werden korrespondierend die drei Design-Räume zugeordnet, die jeweils strukturierte und assoziative Elemente enthalten:

Der *Konsensraum* umfasst die Aushandlung temporärer Konsenszustände. In diesem Raum werden sowohl Dekompositionen (Struktur) vorgenommen als auch Kontext oder Interessen (Assoziation) berücksichtigt. Das Ergebnis der Verhandlungen sind funktionale Spezifikationen, die zusätzlich zu technischen Aspekten auch assoziative Elemente enthalten.

Der *Aufgabenraum* umfasst das Prototyping. In dieser Phase werden unter Einbeziehung der Nutzer Prototypen entwickelt, die sukzessive an Kontext und Benutzerpräferenzen angepasst werden. Als Ergebnis dieses Raums erhält man ein Teilprodukt, das als anpassbares Produkt gestaltet ist.

Im *Implementierungsraum* wird die Anpassung durch die Benutzer vorgenommen. Dabei wird auf das Teilprodukt des Aufgabenraums zurückgegriffen. Das fertige Produkt wird jeweils nur durch die Benutzer komplettiert.

Die 3-Design-Spaces-Strategy behebt Schwächen der bisher erläuterten Modelle insbesondere in Bezug auf Benutzerpartizipation. Die Integration assoziativer wie strukturierter Elemente ermöglicht eine ausgewogene Planung zwischen streng formaler und intuitiver Vorgehensweise. Gerade bei Lernumgebungen ist dieser Aspekt positiv zu beurteilen. Die Strategie identifiziert wichtige Designaspekte, jedoch fehlen konkrete Anpassungsmechanismen zur Ableitung von Projektmodellen oder zu individuellen Vorgehensmodellen. Daher ist die 3-Design-Spaces-Strategy als übergeordnete Richtlinie anzusehen.

3.5 Folgerungen

Es zeigt in der Analyse der einzelnen Vorgehensmodelle, dass weder bestehende didaktische noch technologieorientierte Modelle die folgenden Anforderungen an ein umfassendes Vorgehensmodell vollständig erfüllen können:

- Phasenabdeckung der C-, D- und E-Ebene des didaktischen Handelns
- Integration des organisationalen Kontextes
- Kontextuelle Anpassbarkeit
- Qualitätssicherung
- Qualitätskriterien der Softwareentwicklung
- Benutzbarkeit und Sicherheit
- Portabilität, Interoperabilität und Wiederverwendbarkeit
- Unterstützung von Standards
- Gleichwertige Berücksichtigung didaktischer und technologischer Faktoren
- Werkzeugunterstützung

Die Anforderungen werden an dieser Stelle nicht weiter erläutert, da sie in Abschnitt 5.1 in Bezug auf die Entwicklung von ELM diskutiert werden.

4 Standardisierung von Lerntechnologien

In diesem Abschnitt wird diskutiert, welche aktuellen Standardisierungsansätze die Qualität von Lernumgebungen fördern. Der Fokus dieser Ansätze liegt auf der Austauschbarkeit, Rekombinierbarkeit und Wiederverwendbarkeit der Komponenten computerunterstützter Lernumgebungen. Das bedeutet, dass Lernumgebungen unabhängig von der verwendeten Systemumgebung, dem verwendeten Autorensystem oder dem Kontext eingesetzt werden können. Eine solche Wiederverwendbarkeit ist für die wirtschaftliche Entwicklung qualitativ hochwertiger Lernumgebungen unerlässlich.

Es werden Ansätze unterschiedlicher Forschungs- und Entwicklungsgruppen betrachtet, um eine Bewertung der Ansätze vornehmen zu können. Zum Darstellungsformat sei erwähnt, dass die Bezeichnungen der Standards in englischer Sprache belassen werden, da noch keine international anerkannte Übersetzung vorliegt. Die Erläuterungen werden dagegen in deutscher Sprache vorgenommen.

Abschließend wird gefolgert, welche Standards bzw. Standardisierungsbestrebungen in einem Vorgehensmodell berücksichtigt werden müssen. Dieses Kapitel basiert auf den Ausführungen in [PaAd2001a] und erweitert diese Inhalte um aktuelle Entwicklungen.

4.1 Zielsetzung von Standards

Unter einem *Standard* versteht man die Vereinheitlichung und Formalisierung von Produkten, Diensten und Prozessen in Form von Regeln, Leitlinien oder Spezifikationen. Ein Standard soll der Vereinfachung und Vereinheitlichung von Entwicklungs- und Anwendungsprozessen dienen. Der Begriff Standard wird verwendet, sobald es sich noch nicht um ein formal anerkanntes Dokument der Normungsinstitutionen (z. B. Deutsches Institut für Normung e.V., DIN; International Organization for Standardization, ISO) handelt. Für formal anerkannte Dokumente (z. B. ISO 9000) wird der Begriff *Norm* verwendet. Im Folgenden werden die Problembereiche kurz zusammengefasst, die bei der Standardisierung von Lerntechnologien zu betrachten sind.

Standards werden dabei auf den unterschiedlichen Architekturebenen spezifiziert und können somit nach der Architekturebene (siehe Abschnitt 5.3) klassifiziert werden (Interface, Systeme, Protokolle, Komponenten).

Standards werden in allen Phasen der Verwendung von Lernumgebungen benötigt, also von der Entwicklung bis zur dauerhaften Nutzung, Administration und Wartung. Ferner bieten Standards Vorteile für Entwickler, für Anwender und auch für Organisationen, die Lernumgebungen für Mitarbeiter oder Kunden zur Verfügung stellen: Durch die Vereinheitlichung von Datenaustausch, Produkt- oder Prozessbeschreibungen wird die Interoperabilität von Systemen oder Komponenten sichergestellt und somit ein aufwendiger Transfer vermieden. Dabei müssen Standards verschiedene Anforderungen erfüllen, um sinnvoll einsetzbar zu sein:

- *Rekombinierbarkeit*: Lerninhalte, Methoden und Spezifikationen müssen in unterschiedlichen Systemumgebungen und Lernsituationen wiederverwendbar sein.

- *Rekontextualisierung*: Lerninhalte müssen in unterschiedlichen Kontexten verwendbar sein. So sollten zum Beispiel ausgewählte Inhalte von universitären Veranstaltungen auch für berufliche Weiterbildungsmaßnahmen genutzt werden können.
- *Portabilität/Interoperabilität*: Ein Standard für Lerntechnologien muss system-, plattform und anwendungsunabhängig sein.
- *Adaptation/Adaptierbarkeit*: Lerntechnologien müssen an verschiedene Lernsituationen (unterschiedliche Benutzer, unterschiedlicher Anwendungskontext) anpassbar sein.
- *Flexibilität*: Ein Standard für Lerntechnologien darf Lehrende, Lernende und Entwickler nicht ihrer Gestaltungsfreiheit beim didaktischen wie technologischen Design einer Lernumgebung einschränken. Es ist offensichtlich, dass gerade der Entwicklungsprozess von Lernumgebungen ein hohes Maß an Intuition, Kreativität und Spontaneität einschließt.
- *Einfachheit*: Um Entwickler und insbesondere eher didaktisch als technisch interessierte Ausbilder oder Lehrende in den Spezifikationsprozess von Standards einzubeziehen, müssen standardkonforme Spezifikationen einfach und mit vertretbarem Mehraufwand zu erstellen sein. Dabei muss der Nutzen der Standards deutlich sein, um die Akzeptanz und Motivation zur Mitarbeit aller Akteure an Standardspezifikationen sicherzustellen.
- *Wirtschaftlichkeit*: Ein Standard muss zu einer verbesserten Wirtschaftlichkeit der Entwicklung und Anwendung von Lerntechnologien führen.

Die Akzeptanz und damit der Erfolg der Standardisierungsbestrebungen hängt wesentlich von der Erfüllung dieser Kriterien ab. Im Folgenden werden die jeweiligen Entwicklungen bezüglich dieser Kriterien betrachtet.

4.2 Standardisierung von Lerntechnologien

Die Vielzahl von Standardisierungspotenzialen erfordert eine weitere Eingrenzung des Arbeitsgebiets, da gerade im Kontext des CUL verschiedenste Technologien (z. B. Videoconferencing oder Streaming) genutzt werden, die ebenfalls auf Standards (z. B. Standardisierung der Protokolle) basieren. Es lassen sich zwei Tätigkeitsfelder von Standardisierungsaktivitäten für Lerntechnologien unterscheiden:

- *Spezifische Standards für Lerntechnologien* (siehe 4.4) sind Standards, die im unmittelbaren Umfeld computerunterstützter Lernumgebungen anzusiedeln sind. Die Zielsetzung dieser Initiativen ist die Sicherstellung der Interoperabilität und Wiederverwendbarkeit von Lernumgebungen für Lehrende, Lernende und Entwickler.
- *Korrespondierende Standards* (siehe 4.2.1) hängen nur mittelbar mit dem Bereich der Lerntechnologien zusammen und wurden für andere Zwecke entwickelt. Sie werden im Rahmen der spezifischen Standards eingesetzt. So verwenden zum Beispiel die spezifischen Standards des *Instructional Management Systems*-Projekt die Auszeichnungs-

sprache bzw. den korrespondierenden Standard *Extensible Markup Language (XML)* zur Spezifikation ihrer Entwicklungen. Maßgebliche Standards sind in diesem Zusammenhang zum Beispiel *Resource Description Framework (RDF)*, *Dublin Core (DC)* oder Standards im Bereich des Electronic Business. Weitere technologische Standards (z. B. Protokolle) werden als Grundlage vorausgesetzt und in dieser Arbeit nicht näher erläutert; XML wird dagegen aufgrund der Bedeutung (z. B. für die Implementierung der Lerntechnologiestandards) näher beschrieben. Weiterhin werden in diesem Bereich Standards erläutert, die mit Lern- und Entwicklungsprozessen mittelbar zusammenhängen: So werden *administrative Standards* zum Beispiel in Hochschulverwaltungen eingesetzt; *E-Business-Standards* können zum Beispiel für den Handel mit Lernumgebungen eingesetzt werden.

4.2.1 Korrespondierende Standards

Lerntechnologiestandards, die Metadaten (siehe 4.4.3) einbeziehen, basieren auf dem Dublin Core-Standard (siehe 4.2.1.1) zur Beschreibung von Metadaten für elektronische Ressourcen. Dieser Standard wurde entwickelt, um eine allgemeingültige formale Repräsentation von Metadaten zur Verfügung zu stellen und dient als Basis für spezialisierte Metadaten.

Des Weiteren greift man derzeit bei der Implementierung der Lerntechnologiestandards auf verschiedene technologische Standards wie XML [WWWC2000], HTML [WWWC1999a] oder RDF [WWWC1999b] zurück, um allgemeingültige, austauschbare Formate zu verwenden. Auch dieser Standards befinden sich derzeit noch in der Entwicklung, daher müssen die entsprechenden Implementierungen auch in der Zukunft weiter angepasst werden. Die Standards sind als Grundlage für diese Arbeit zu sehen und werden daher nicht weiter betrachtet. Da Lernumgebungen im Essener-Lern-Modell unter Verwendung von XML generiert werden und die Spezifikationen der Informationsmodelle der Lerntechnologiestandards in XML spezifiziert sind, wird dieser Standard in Abschnitt 4.3 separat erläutert.

4.2.1.1 Dublin Core Metadata-Initiative

Die Dublin Core (DC) Metadata-Initiative definiert eine Menge von 15 Elementen zur Beschreibung von Ressourcen. Unter einer Ressource versteht man dabei ein Objekt mit einer Identität, typischerweise Informations- oder Servicere Ressourcen [DCMI1999]. Diese 15 Elemente bilden eine minimale Menge an Informationen, um eine Ressource eindeutig zu beschreiben. Qualifier ergänzen die Elemente um weitere Beschreibungen [CEN2000b].

Tabelle 16: Dublin Core-Metadaten

| Element | Definition | Element refinements |
|--------------|---|--|
| Title | Name der Ressource | Alternative |
| Creator | Entity, verantwortlich für den Inhalt einer Ressource | - |
| Subject | Inhalt einer Ressource | - |
| Description | Beschreibung des Inhalts einer Ressource | Table of Contents, Abstract |
| Publisher | Entity, verantwortlich für die Veröffentlichung einer Ressource | - |
| Contributor | Entity, verantwortlich für Beiträge zu einer Ressource | - |
| Date | Datum eines Ereignisses im Lebenszyklus einer Ressource | Created, Valid, Available, Issued, Modified |
| Type | Inhaltsbereich einer Ressource | - |
| Format | Physische oder digitale Manifestation einer Ressource | Medium, Extent |
| Identifizier | Eindeutige Referenz auf eine Ressource innerhalb des Kontexts | - |
| Source | Quelle, von der eine Ressource abgeleitet ist | - |
| Language | Sprache der Ressource | - |
| Relation | Referenz zu verwandter Ressource | Is Version Of, Has Version, Is Replaced By, Replaces, Is Required By, Requires, Is Part Of, Has Part, Is Referenced By, References, Is Format Of, Has Format |
| Coverage | Bereich, der von einer Ressource abgedeckt wird | Spatial, Temporal |
| Rights | Rechte zu einer Ressource | - |

DC-Metadaten werden in verschiedenen Anwendungsbereichen eingesetzt. Insbesondere Bibliotheken verwenden das Modell, um Ressourcen zu organisieren und interoperabel verwenden zu können. Derzeit arbeitet eine Arbeitsgruppe (DC Education) innerhalb der DC Metadata-Initiative an einer Erweiterung der Metadaten um den Bildungskontext. Dabei werden folgenden Datenelemente der Learning Object Metadata (siehe 4.4.3.1) zur Erweiterung empfohlen [Duva2001]:

- *Audience* beschreibt die Rolle, für die eine Ressource vorgesehen ist (z. B. Lehrender, Autor, Endbenutzer, Lernender).
- *Educational Standards* beschreiben organisationale, regionale, nationale und internationale Standards, die einer Ressource zugrunde liegen.

- *Interactivity type*, *interactivity level* and *typical learning time* sind äquivalent zu den entsprechenden LOM-Elementen.

Die Akzeptanz und Verbreitung des DC Standards hat zur Folge, dass dieses Format auch für Ressourcen im Bildungsbereich verwendet werden sollte. Dennoch können bildungsspezifische Ressourcen durch die Kernelemente nicht hinreichend exakt beschrieben werden. Auch die Erweiterung ermöglicht noch keine detaillierte Einordnung einer Bildungsressource. Daher ist die Beschreibung durch LOM, die die DC-Elemente als Teilmenge enthalten, weiterhin sinnvoll.

4.2.1.2 E-Business Standards

Die Nutzung des Internets als Informations-, Kommunikations- und Integrationsplattform hat auch im Bildungsbereich wesentlich zugenommen. Die Unterstützung von Geschäftsprozessen durch Informations- und Kommunikationstechnologien werden dabei unter dem Begriff *Electronic Business* zusammenfasst [vgl. PaAd2001b]. So werden Lernumgebungen über Portale vertrieben und genutzt (z. B. Virtuelle Universitäten [vgl. Schu2001]), Entwickler finden Unterstützungsfunktionen in Virtuellen Communities. Ein wichtiges Anwendungsfeld computerunterstützter Lernumgebungen werden insbesondere *Electronic Education Markets (EEM)* bilden. Elektronische Märkte sind interorganisationale Informationssysteme, die Funktionen zur Information, Kommunikation und Integration von Prozessen bieten. Electronic Education Markets sind darauf aufbauend als Systeme des Electronic Business im Bildungsbereich anzusehen. Es ist abzusehen, dass ein breiter Markt für Lernumgebungen außerhalb klassischer Bildungseinrichtungen entstehen wird [vgl. LaMM2001] und somit die Entwicklung, die Verwendung und der Handel über EEM abgewickelt werden. Dabei müssen Schnittstellen zu betriebswirtschaftlichen Anwendungen geschaffen werden, um beteiligte Systeme integrieren zu können. Um diese Integration zu vereinfachen, entwickeln verschiedene Initiativen Standards, die den Informationsaustausch und Integration erleichtern.

Beispielhaft sei die *Electronic Business XML (ebXML)* genannt [EBXM2000]. Zielsetzung der ebXML ist es, einen Standard zum Datenaustausch durch elektronischen Dokumente in verschiedensten Geschäftsprozessen zu entwickeln. Dazu werden folgende Bereiche abgegrenzt:

- Es werden generische *Geschäftsprozesse* in Form eines Metamodells definiert, auf die organisationsspezifische Prozesse abgebildet werden können. Eine Übersicht über die identifizierten Prozesse wird in Tabelle 17 gezeigt.
- Weiterhin werden *Dokumente* spezifiziert, die in diesen Prozessen ausgetauscht werden können.
- Es werden *Protokolle* spezifiziert, die den Transfer von elektronischen Dokumenten regeln.

Tabelle 17: Geschäftsprozesse innerhalb von ebXML [EBXM2000]

| Bereich | Prozesse |
|-------------------------------------|--|
| Administration | Failure notification, product service review, party profile, party agreement |
| Procurement Management | Pre-Order and Quote, Payment Term Negotiation, Forecast/Planning, Procurement, Authorization, Transportation/Distribution, Return of Goods, Payment, Product Configuration |
| Product Introduction | Preparation for Distribution, Product Change Notification |
| Inventory Management | Inventory Reporting, Sales Reporting |
| Financial | Invoicing, Payroll |
| International Transport and Customs | Declaration of Goods, Inbound Transport, Outbound Transport, Operation, Clearance, Cross Border Return Handling |
| Marketing Information Management | Lead Opportunity Management, Marketing Campaign Management, Design Win Management |
| Service and Support | Warranties, Service Packages, Contracted Service, Asset Management |

Durch die Verwendung dieses Standards ist es möglich, betriebliche Prozesse zu integrieren und somit den Handel von Lernumgebungen zu vereinfachen. Weitere Standards werden derzeit für die Erstellung von Produktkatalogen (Commerce XML, CXML [CXML2000] und BMEcat [HüRS1999, HüSc2001, BMEc2001]) oder in verschiedenen Teilbereichen (insbesondere Abrechnungssysteme wie zum Beispiel Electronic Commerce Markup Language ECML [ECML1999]) entwickelt. Da dieser Bereich außerhalb der Zielsetzung dieser Arbeit liegt, sei auf [BiPV2001, PaAd2001b] verwiesen.

4.2.1.3 Administrative Standards

Administrative Standards werden im Bereich der Organisation und Administration von Bildungseinrichtungen verwendet. Abläufe in Bildungseinrichtungen sollen vereinfacht und Kollaboration zwischen verschiedenen Institutionen (z. B. im Rahmen von Kooperationen oder virtuellen Organisationen) soll gefördert werden. Ein Standard zu diesem Zweck ist die *Schools Interoperability Framework Implementation Specification (SIF)*. Diese Spezifikation betrachtet den Datenaustausch in folgenden Bereichen: Infrastructure, Exceptionalities, Food Services, HR/Financials, Instructional Management, Library, Student Information, Transportation/Geographic Information. Auch wenn dieser Standard eine andere Perspektive auf Bildungsorganisation einnimmt, so müssen doch gerade Spezifikation im Bereich der Prüfungsverwaltung, Studentenadministration und Studenteninformatio

diniert werden. Dennoch liegt dieser Standard nicht im Geltungsbereich dieser Arbeit, für weitere Informationen sei auf [SIIA2000] verwiesen.

4.2.2 Spezifische Standards

Derzeit gibt es verschiedene Aktivitäten mit unterschiedlichen Schwerpunkten im Bereich der Lerntechnologiestandards. Abbildung 36 gibt eine Übersicht über die Initiativen und Forschungsprojekte, die in diesem Artikel vorgestellt werden, wobei der Fokus auf spezifischen Standards liegt.

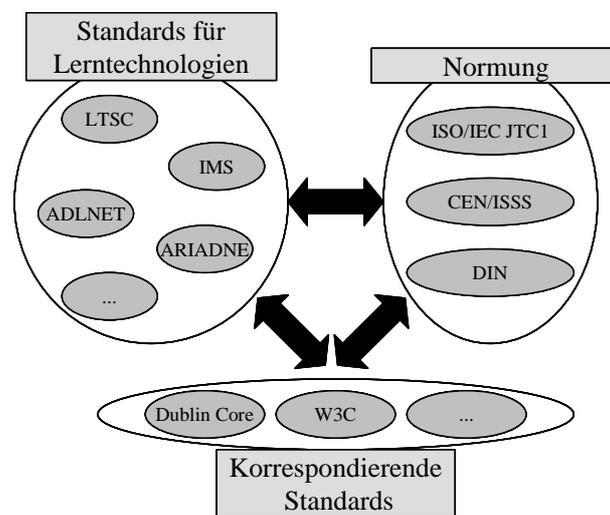


Abbildung 36: Zusammenhang der Standardisierungsaktivitäten

Spezifische Lerntechnologiestandards werden insbesondere vom *Learning Technology Standards Committee (LTSC)* der IEEE erarbeitet. Ziel der Arbeitsgruppen ist die Entwicklung technischer Spezifikationen, Empfehlungen und Richtlinien für Softwarekomponenten, Tools, Technologien und Methoden. Dabei liegt der Fokus auf der Entwicklung technischer Spezifikationen. Weitere Standards (z. B. Bildungsstandards, Managementstandards, Evaluation) liegen außerhalb des Wirkungsbereichs der LTSC. Derartige Standards sollen die Entwicklung, Umsetzung, Wartung und Interoperabilität von Lernsystemen unterstützen. Ausgehend von einer Systemarchitektur, der *Learning Technology Systems Architecture (L TSA)*, die bereits in Abschnitt 2.4.2 erläutert wurde, werden Standards für verschiedene Teilbereiche entwickelt. Neben der Architektur existieren derzeit zwei Standards mit einem Reifegrad, der erwarten lässt, dass sie innerhalb der nächsten Jahre in eine Norm überführt oder zumindest allgemein akzeptiert und verwendet werden:

- *Learning Object Metadata (LOM)* beschreiben die Eigenschaften von Lernressourcen.

- *Computer Managed Instruction (CMI)* unterstützt die Ablaufsteuerung von Lernprozessen durch Lernmanagementsysteme und die Interoperabilität verschiedener Systeme.

Die Aktivitäten der LTSC sind als abstrakt-konzeptuell zu beurteilen. Daneben ist das *Instructional Management Systems (IMS)*-Projekt zu nennen. In diesem Konsortium aus Unternehmen, Bildungsinstitutionen und Forschungseinrichtungen werden in enger Zusammenarbeit mit der LTSC und weiteren Standardisierungsinitiativen ebenfalls Spezifikationen zur Interoperabilität von Lerntechnologien entwickelt. Die Standards des IMS-Projekts sind als praxis- und anwendungsorientiert zu bewerten. Diese orientieren sich größtenteils an den Modellen der LTSC, jedoch werden hier Lösungen zur Implementierung vorgegeben und detaillierte Vorgehensrichtlinien für diesen Bereich erstellt.

Ergänzend sei auf die Initiative *Advanced Distributed Learning (ADL)*, die 1997 vom amerikanischen Department of Defense (DoD) und dem White House Office of Science and Technology Policy (OSTP) gegründet wurde, hingewiesen. Diese Initiative versucht, verschiedene Ansätze zu einem gemeinsamen Rahmenkonzept zu integrieren. So sind hier die maßgeblichen Akteure der Lerntechnologiestandardisierung vertreten. Beispielsweise entwickelt das amerikanische Aviation Industry CBT Committee (AICC) Standards und zertifiziert Produkte und Dienstleistungen von Ausbildungsanbietern der Luftfahrtindustrie. Weiterhin sind hier Vertreter der LTSC, IMS oder ARIADNE (Alliance of Remote Instructional Authoring Distribution Networks for Europe) organisiert. In dieser Gruppe wurde, basierend auf den Aktivitäten der LTSC, AICC und IMS, das Sharable Content Object Reference Model (SCORM, siehe 4.5) entwickelt, das Standards verschiedener Ebenen integriert [Dodd2001a].

Ferner gibt es eine Vielzahl an Projekten, die sich mit der Umsetzung, der Implementierung und der Entwicklung von Tools zur Verwendung von Standards befassen. Im europäischen Raum zählt dazu in erster Linie das Projekt ARIADNE (Alliance of Remote Instructional Authoring Distribution Networks for Europe). Ziel dieses Projekts ist es, einen europäischen Wissenspool computerunterstützter Lernressourcen aufzubauen und entsprechende Entwicklungswerkzeuge zur Verfügung zu stellen [ARIA2000a, DDVC2001]. Dabei werden nicht nur Standards genutzt und implementiert, sondern auch Entwicklungs- und Forschungsarbeit geleistet.

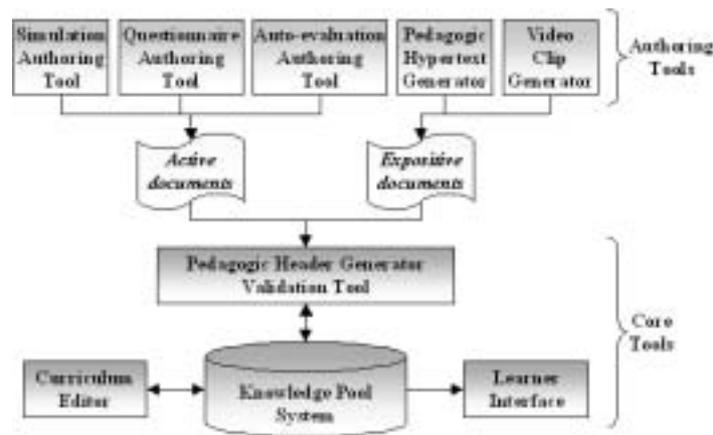


Abbildung 37: Übersicht der Tools im Projekt ARIADNE [ARIA2000b]

Das ARIADNE-Projekt kooperiert intensiv mit zahlreichen Standardisierungsinitiativen (z. B. LTSC, CEN). Aufgrund der resultierenden Synergieeffekte werden sich die Arbeitsergebnisse auch in zukünftigen Entwicklungen dieser Initiativen niederschlagen. Insbesondere Universitäten können europaweit auf den *Knowledge Pool* zugreifen und im Gegenzug Lernressourcen zur Verfügung stellen. Dennoch ist die Akzeptanz und damit die Nutzung der Tools besonders im deutschsprachigen Raum derzeit noch nicht verbreitet.

Aus den bisher beschriebenen Standards gehen verschiedene *Normierungsbestrebungen* hervor. Insbesondere aus den Standardisierungsaktivitäten der LTSC und der IMS soll eine Norm der International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) entstehen.

Auf europäischer Ebene befasst sich das *CEN/ISSS (European Committee for Standardization/Information Society Standardization System)* im Workshop *Learning Technologies* mit der formalen Normung. Dabei fließen die Ergebnisse der oben genannten Aktivitäten in den Normungsprozess ein. Derzeit wurden zwölf Empfehlungen gegeben [CENI2000a], die in Normen münden sollen. Diese beziehen sich auf die Teilbereiche:

- *Metadaten:* In diesem Zusammenhang werden, neben den bereits bestehenden Metadaten, Taxonomien entwickelt, die die Einordnung von Lerntechnologien ermöglichen. Dabei werden auch Implementierungen als Richtlinien in verschiedenen Formaten (XML, RDF) zur Verfügung gestellt.
- *Multilingualität und Internationalisierung:* Die Normen, Standards und Implementierungen sollen aufgrund des europäischen Kontextes internationalisiert werden.
- *Rechtliche Normen:* Die Normen müssen aufgrund der unterschiedlichen nationalen Rechtslagen, zum Beispiel bezüglich des Datenschutzes und Urheberrechts, angepasst werden.
- *Kollaboratives Lernen:* Normen in diesem Bereich unterstützen Kooperation und Kommunikation in verteilten Lernsituationen.
- *Lernermodelle:* Neben persönlichen Basisinformationen werden Daten über Aktoren von Lernprozessen erfasst und ausgetauscht.

- *Qualitätssicherung*: In den bisher genannten Spezifikationen wird der Aspekt der Qualitätssicherung nicht explizit diskutiert. In diesem Rahmen wird Standard zur Beurteilung und Zertifizierung von Lernressourcen entworfen.

4.3 Extensible Markup Language (XML)

In diesem Abschnitt erfolgt eine kurze Einführung in die Extensible Markup Language (XML) und deren kooperierende Standards. Insbesondere wird hier auf die Dokumenttyp-Definition (DTD), die Extensible Stylesheet Language (XSL) und Extensible Linking Language (XLink) eingegangen.

4.3.1 XML-Konzept

Die *Extensible Markup Language (XML)* wurde im November 1996 als Entwurf vorgestellt und hat im August 2000 als XML 1.0 (Version 2) den Status einer World Wide Web Consortium (W3C)-Empfehlung [WWWC2000] erreicht. Es handelt sich hierbei derzeit noch nicht um eine Norm. Die Entwicklung entstand einerseits vor dem Hintergrund, den zu erwartenden proprietären Entwicklungen von HTML entgegenzuwirken, andererseits wurde ersichtlich, dass der Einsatz der Standard Generalized Markup Language (SGML) aufgrund ihrer Komplexität nicht realisierbar ist. „Als Kompromiss zwischen Einfachheit (HTML) und Funktionsvielfalt (SGML) ist XML eine konsequent reduzierte Version von SGML [...], das bei vertretbarem Aufwand maximale Flexibilität in der Definition von Datenformaten (im Sinne von Auszeichnungssprachen) [...] garantiert. [...] Die beiden wichtigsten korrespondierenden Sprachen von XML, XSL und XLL, waren von Beginn an als komplementäre Teile von XML geplant.“ [Mich1999]

Das große gegenwärtige Interesse an XML begründet sich darin, dass neben den vielfältigen Möglichkeiten der dynamischen Repräsentation von Inhalten, Daten und Informationen eine Technologie zur Verfügung steht, die, ähnlich mächtig wie SGML, universelle Möglichkeiten zum Austausch, zum Retrieval und zur Verwaltung semantisch qualifizierter Daten unterstützt. XML unterstützt die Kompatibilität von Daten in beliebigen Verwendungszusammenhängen (z. B. Datenbanken), ermöglicht eine intelligente kontextuelle Recherche sowie die Mehrfachverwendung von Informationen.

XML ist jedoch keine direkte Auszeichnungssprache wie HTML, sondern vielmehr eine Metasprache, die es erlaubt, eigene Dokumenten-Markups zu schaffen und ein Dokument damit zu strukturieren. XML umfasst kein eigenes Vokabular zur Auszeichnung von Dokumenteninhalten, es handelt sich hierbei um eine formale Sprache zur Erzeugung (Konstruktion/Definition) von Auszeichnungssprachen. In diesem Zusammenhang stellt HTML bzw. die XML-kompatible Version von HTML die sogenannte Extensible Hypertext Markup Language (XHTML) eine der Auszeichnungssprachen dar, die mit der Metasprache XML definiert werden können. Zusammenfassend lassen sich die Vorteile der Verwendung einer generischen, deskriptiven Metaauszeichnungssprache in drei Gruppen kategorisieren:

- Austauschbarkeit
- Modularisierung
- Qualitätssicherung

Die *Austauschbarkeit* beschreibt insbesondere die durch XML unterstützte Trennung von Struktur, Format und Inhalt eines XML-Dokumentes. Mithilfe deskriptiver Metainformationen wird ein gezieltes inhaltliches *Modularisieren* von Dokumenten unterstützt, d. h. einzelne Dokumente bzw. Teile können angesteuert, kopiert, gefiltert, aktualisiert oder referenziert werden.

Die mit XML zu definierende Struktur von Dokumententypen stellt dem Verfasser eines XML-Dokumentes ein Modell für den inhaltlichen Aufbau zur Verfügung. Daraus resultiert eine Strukturvorgabe, die einerseits als Standard dient, andererseits aber auch den Autor dahingehend unterstützt, wesentliche Inhalte nicht auszulassen. Dies hat einen Anstieg der *Qualität* von XML-Dokumenten zur Folge. Voraussetzung hierfür sind jedoch standardisierte DTD, da die Qualität eines XML-Dokumentes in direktem Bezug zu der zugehörigen DTD steht.

4.3.2 Technologie

Seit dem ersten XML-Entwurf im November 1996 sind ergänzende sowie domänen- und kontextspezifische Spezifikationen entwickelt worden. Diese werden häufig synonym zum Begriff XML verwendet. Dies ist auch im Rahmen der vorliegenden Arbeit der Fall: die XML-basierte Lernumgebung verwendet sowohl die Grundkonzepte der Markup-Sprache als auch kooperierende Standards wie XSL und XLink. Kooperierende Standards sind überwiegend Sprachen, die zur Definition anderer Sprachen verwendet werden, wohingegen XML-Applikationen direkte Anwendungen zur Auszeichnung von Dokumenten darstellen [Mich1999].

Eine explizite Beschreibung aller dargestellten kooperierenden Standards und XML-Anwendungen würde an dieser Stelle zu weit führen, dennoch sollen in Tabelle 18 zumindest die Akronyme aufgelöst werden:

Tabelle 18: Zusammenfassung kooperierender Standards und Applikationen

| Kooperierender Standard | Beschreibung |
|--------------------------------------|---|
| XSL (Extensible Stylesheet Language) | Sprache zur Spezifikation der Formatierung |
| XLink (Extensible Linking Language) | beschreibt Verweise in XML-Dokumenten |
| XML DOM (XML Document Object Model) | definiert den Zugriff auf die Inhalte von XML-Dokumenten |
| RDF (Resource Description Format) | Formulierung und Verarbeitung von Metadaten zur Beschreibung von Ressourcen |

| XML-Applikationen | Beschreibung |
|---|---|
| XQL (XML Query Language) | XML-basierte Abfragesprache |
| JSML (Java Speech Markup Language) | Aufbereitung von Text-Eingaben für Sprach-Synthesizer |
| XHTML (Extensible HTML) | XML-kompatible Version von HTML |
| VXML (Voice eXtensible Markup Language) | Codierung lautsprachlicher Informationen |
| XML EDI (XML Electronic Data Interchange) | Austausch von XML-Dokumenten per EDI |
| MathML (Mathematical Markup Language) | Codierung mathematischer Ausdrücke in XML-Dokumenten |
| SMIL (Synchronized Multimedia Integration Language) | Beschreibung von multimedialen Inhalten |
| SpeechML (Speech Markup Language) | Konstruktion netzwerkbasierter auditiver Dokumente |

4.3.3 Konzepte

Nachdem zuvor die Rahmenbedingungen der XML-Technologie dargestellt wurden, wird im Folgenden einführend die XML-Terminologie vorgestellt. Diese Betrachtungen basieren auf [W3C2000, GoPr1999, Mich1999, PoWi2000]. Zur Bearbeitung von XML-Inhalten werden im Allgemeinen drei Konzepte benötigt:

- XML-Dokument
- Dokumenttypdefinition (DTD)
- Stylesheet

Inhalt des *XML-Dokuments* sind der Prolog und die Dokumentdaten, welche durch aussagekräftige XML-Elemente gekennzeichnet sind. Ein XML-Dokument besteht aus einem oder mehreren *Elementen*. Abbildung 38 zeigt die Kennzeichnung eines XML-Elementes:

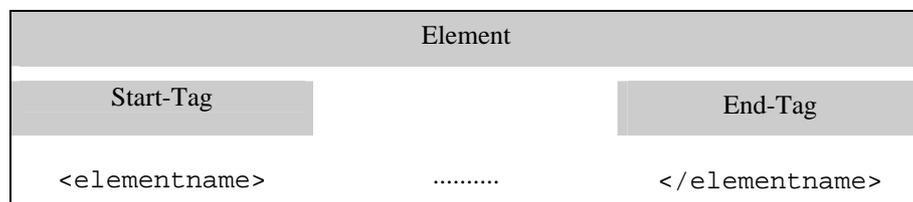


Abbildung 38: Kennzeichnung eines XML-Elementes

Die *Dokumenttypdefinition (DTD)* bestimmt die Regeln, wie XML-Elemente, Attribute und andere Daten definiert, (physisch) strukturiert und mit einem logischen Bezug zueinander in einem XML-Dokument dargestellt werden können. Ziel der DTD ist die Strukturierung von Inhalten für einen bestimmten Dokumenttyp. Innerhalb einer DTD werden die Beziehungen zwischen den Inhalten, die charakteristisch für alle Dokumente eines Typs

sind, festgelegt. Zwischen Dokumenttyp und Dokument besteht eine normative Beziehung: Die DTD schreibt vor, wie ein Dokument des betreffenden Typs auszuzeichnen ist. In einer DTD werden somit alle Auszeichnungen sowie die zugehörigen Attribute definiert. Es wird festgelegt, ob Auszeichnungen verschachtelt auftreten dürfen und gegebenenfalls welche Auszeichnungen innerhalb anderer angewandt werden dürfen. Die Qualität von XML-Dokumenten lässt sich so wesentlich steigern.

Stylesheets beschreiben, wie Elemente formatiert dargestellt werden sollen. Dabei können verschiedene Stylesheets auf das gleiche Dokument angewendet werden, d. h., dessen Erscheinungsbild kann verändert werden, ohne die zu Grunde liegenden Daten zu beeinflussen. Die Trennung zwischen Inhalt und Formatierung ist ein wichtiges Merkmal in XML. Im Zusammenhang mit der Lernumgebungen werden Stylesheets insbesondere zur individuellen Anpassung von Präsentationsinhalten verwendet.

XSL ist eine XML-Sprache zur Spezifikation der Formatierung und Präsentation von XML-Dokumenten. Für eine detaillierte Betrachtung von XSL müssen drei Alternativen berücksichtigt werden, da die XSL-Spezifikation derzeit noch nicht abgeschlossen ist. Im Rahmen des W3C-Standardisierungsverfahrens handelt es sich nur um einen Working Draft. Es können folgende Alternativen bzw. sinnverwandte Konzepte von XSL verwendet werden:

- Die *Document Style Semantics and Specification Language (DSSSL)* liegt dem Standard XSL zu Grunde. DSSSL beschreibt die Stilsprache, die im Zusammenhang mit SGML entwickelt wurde. Ähnlich wie SGML ist zwar auch DSSSL flexibel, jedoch sehr umfangreich und schwer handhabbar.
- *Cascading Style Sheets (CSS)*: Die Verwendung von CSS im Zusammenhang mit XML sollte nur ergänzend erfolgen, da CSS zwar für ein starres Modell wie HTML anwendbar sind, jedoch den Anforderungen von XML nicht gerecht werden.
- *Extensible Stylesheet Language Transformations (XSLT)* beschreibt im engeren Sinne eine unabhängige Sprache zur Transformation von XML-Dokumenten in XML-Dokumente mit anderer Struktur und anderem Vokabular, d. h., es ist eine Sprache zur Formulierung von system- und anwendungsunabhängigen Stylesheets. Im weiteren Sinne werden XSLT und XSL häufig synonym verwendet. Die Möglichkeiten von XSLT sind gegenüber XSL zwar geringer; Vorteil von XSLT ist jedoch, dass diese Spezifikation bereits seit zwei Jahren den Status einer W3C-Empfehlung hat [WWWC1999c], während XSL erst im Oktober 2001 diesen Status erreicht hat [WWWC2001b]. Dementsprechend wird XSLT bereits wesentlich besser unterstützt (z. B. Internet Explorer ab Version 5.0 als Browser, entsprechende Werkzeuge). Die in ELM verwendeten Stylesheets beziehen sich aufgrund der derzeitigen Anwendbarkeit im Wesentlichen auf die XSLT-Spezifikation.

Zusätzlich besteht die Möglichkeit der Abbildung einer Hypertextstruktur durch die Verwendung von XLink, einer weiteren Komplementärsprache [WWWC2001a]. Im Gegensatz zu HTML hat XML keine direkten Elemente für Hyperlinks; dieses Defizit wird durch XLink aufgehoben. Ziel der Entwicklung von XLink ist es, die Defizite des

HTML/XHTML-Verweiskonzeptes durch eine Neukonzeption zu korrigieren. Es wird *einfache* und *erweiterte Verweise* unterschieden.

Neben diesem Verweiskonzept enthält XLink das sogenannte XPointer Konzept zur Identifikation einzelner Teile von XML-Dokumenten, so dass auf Knoten bzw. Elemente verwiesen werden kann.

Die *einfachen Verweise* sind vergleichbar mit dem HTML/HTTP Verweiskonzept:

```
<element xml:link="simple" href="uri"> ... </element>
```

Weiterhin existieren *erweiterte Verweise*. Mithilfe von erweiterten Verweisen werden folgende Ziele verfolgt:

- *Kommentarfunktion*: Hinzufügen von Kommentaren zu den Verweisen einer Webseite.
- *Multipler Verweis*: Verknüpfung eines Objekts mit n Objekten.
- *Verweisgruppen*: Liste von Verweisen auf XML-Dokumente.

4.4 Lerntechnologiestandards

Im Folgenden werden die sich derzeit in der Entwicklung befindlichen Lerntechnologiestandards beschrieben. Dabei werden die einzelnen Standards auf ihre Einsetzbarkeit hin untersucht.

4.4.1 Kategorisierung und Spezifikationsformate

Standards lassen sich nach verschiedenen Unterscheidungsmerkmalen klassifizieren. So unterteilt das LTSC Aktivitäten in fünf Kategorien:

- *Generelle Standards* (siehe 4.4.2)
- *Metadaten* (siehe 4.4.3)
- *Lernorientierte Standards* (siehe 4.4.4)
- *Inhaltsorientierte Standards* (siehe 4.4.5)
- *Managementorientierte Standards* (siehe 4.4.6)

Dieser Klassifikation fehlen didaktische und methodische Standards zur Darstellung von didaktisch orientierten Komponenten. Daher wird diese Kategorie zusätzlich betrachtet.

Die einzelnen Kategorien werden nachfolgend im Detail erläutert. Eine weitere Möglichkeit ist die Einordnung von Standards bezüglich der Architektur aus Abschnitt 5.3. Diese Einordnung wird abschließend vorgenommen, um aufzuzeigen, welche weiteren Standardisierungspotenziale bestehen und in welchen Bereichen weitere Forschungsarbeiten notwendig sind.

Die Standards der LTSC werden jeweils durch ein Informationsmodell im Spezifikationsformat ISO 11404 spezifiziert; damit wird eine einheitliche Spezifikationsprache verwendet. Hinzu kommen die jeweiligen Kodierungen, die eine Umsetzung der Spezifikationen darstellen:

- *Kodierungen*: Implementierung des Informationsmodells (z. B. in XML oder RDF)
- *API*: Steuerungsschnittstelle für den Datenaustausch (z. B. JavaScript, C++)
- *Protokolle*: Übertragungsprotokolle für einen Standard (z. B. HTTP, CORBA)

Das IMS-Projekt beschreibt zu jeder Spezifikation drei Teilspezifikationen. Das *Information Model* ist das konzeptuelle Modell der Spezifikation. Die *XML Binding Specifications* bilden eine Umsetzung des konzeptuellen Modells in der Auszeichnungssprache XML. Schließlich bietet der *Best Practice and Implementation Guide* einen Leitfaden für die Umsetzung mit entsprechenden Beispielen für Implementierungen.

4.4.2 Generelle Standards

Generelle Standards bilden den Rahmen für die Entwicklung spezifischer Standards. In dieser Kategorie werden derzeit zwei Standards entwickelt. Einerseits wird als Rahmen für weitere Standards eine abstrakte Systemarchitektur von Lerntechnologien spezifiziert (LTSA, siehe 2.4.2). Andererseits entsteht ein *Glossar* [LTSC2001d], das eine Basis für die in diesem Bereich äußerst uneinheitliche Terminologie darstellt. Das Glossar ist derzeit allerdings noch unvollständig und kann noch nicht sinnvoll verwendet werden.

4.4.3 Metadaten

Unter Metadaten versteht man eine minimale Menge an Daten zur Beschreibung von Ressourcen. [Ahro1998] und [GrWa1997] formulieren folgende Anforderungen für die Spezifikation von Metadaten:

- Definition in einem Data Dictionary zur Beschreibung der Elemente, bestehend aus Pflichtelementen und optionalen Elementen
- Verfügbarkeit von Methoden zur Manipulation der Metadaten
- Regeln zur Identifikation, Extrahierung und Erweiterung der Daten
- Verantwortlichkeit einer Organisation
- Internationalisierung der Metadaten
- Verfügbarkeit von Werkzeugen zur Generierung, Übertragung und Speicherung der Metadaten

Eine vielversprechende Entwicklung in diesem Bereich, die bereits in der Praxis verwendet wird [REHW2001, KSHM2001, KoHo2001], ist die Beschreibung von Lernressourcen durch Learning Object Metadata (LOM). Um diesen Standard weltweit verfügbar zu machen, werden die LOM-Spezifikationen in der Arbeitsgruppe *Localization* jeweils landes-

spezifisch übersetzt und angepasst. Weiterhin soll ein Standard für den *Datenaustausch* von Lernumgebungen und Managementsystemen entwickelt werden.

4.4.3.1 Learning Object Metadata (LOM)

Ziel der LOM-Spezifikation ist die Beschreibung digitaler und nicht digitaler Ressourcen, die im Kontext computerunterstützten Lernens genutzt werden [IEEE2000b]. Als Lernobjekte werden alle Ressourcen verstanden, die von computerunterstützten Lernumgebungen verwendet werden. Dies können Kurse, Software-Tools, einzelne Lerneinheiten, multimediale Objekte oder auch menschliche Akteure sein. Die Beschreibung durch ein LOM soll ein Lernobjekt eindeutig identifizieren. Damit wird das Auffinden, die Distribution und die Wiederverwendung von Ressourcen wesentlich vereinfacht. Derzeit besteht ein LOM aus neun Kategorien:

1. Die Kategorie *General* beschreibt generelle Informationen zu einem Lernobjekt. Dazu gehört insbesondere die Einordnung von Lernobjekten durch Schlagworte und Kataloge. Dabei ist ein Katalog als Verzeichnis zu verstehen, das eine detaillierte Einordnung ermöglicht (z. B. Bibliothekskataloge).
2. Die Kategorie *Lifecycle* beschreibt den Zustand eines Lernobjektes bezüglich seines Lebenszyklus. Dabei werden Veränderungen und Versionen dokumentiert.
3. Die Kategorie *Meta-metadata* beschreibt die Metadaten, die zur Beschreibung des Lernobjektes verwendet werden. Dabei ist zu beachten, dass die Metadaten nur als Baselemente anzusehen sind und kontextspezifisch erweitert werden. Diese Erweiterungen werden unter Verwendung eines Kataloges erfasst.
4. Die Kategorie *Technical* beschreibt die technischen Anforderungen und Charakteristika, die zur Nutzung eines Lernobjektes benötigt werden.
5. Die Kategorie *Educational* beschreibt pädagogische Charakteristika einer Ressource. Dabei werden Zielgruppen und rudimentäre Eigenschaften der didaktischen Methoden angegeben.
6. Die Kategorie *Rights* beschreibt rechtliche Bestimmungen für ein Lernobjekt. Dieses kann Bestimmungen zur Verwertung, Nutzung und zum Ankauf umfassen.
7. Die Kategorie *Relation* beschreibt Beziehungen zu anderen Lernobjekten, die als Voraussetzung oder übergeordnete Objekte genutzt werden sollten.
8. Die Kategorie *Annotation* erfasst Kommentare zur Nutzung eines Lernobjektes. Dazu wird jeweils abgebildet, wer die Bemerkungen hinzugefügt hat.
9. Die Kategorie *Classification* beschreibt die Einordnung eines Lernobjektes innerhalb spezifischer Klassifizierungsschemata.

Die Elemente und Attribute sind in Tabelle 19 zusammengefasst.

Tabelle 19: Learning Object Metadata

| Kategorie | Beschreibung | Attribute |
|-----------------------|--|---|
| General | allgemeine Beschreibung einer Lernressource | Identifier, Title, CatalogEntry (Catalog, Entry), Language, Description, Keywords, Coverage, Structure, Aggregation Level |
| Lifecycle | Entwicklungshistorie und aktuelle Version einer Ressource | Version, Status, Contribute (Role, Entity, Date) |
| Meta-Metadata | Beschreibung des Metadaten-satzes | Identifier, Catalog Entry (Catalog, Entry), Contribute (Role, Entity, Date), Metadata Scheme, Language |
| Technical | technische Anforderungen und Merkmale | Format, Size, Location, Requirements (Type, Name, Minimum Version, Maximum Version), Installation Remarks, Other Platform Requirements, Duration |
| Pedagogical | pädagogische Merkmale | Interactivity Type, Learning Resource Type, Interactivity Level, Semantic Density, Intended end user role, Context, Typical Age Range, Difficulty, Typical Learning Time, Description, Language |
| Rights | Urheberrecht, geistiges Eigentum und die Nutzungsbedingungen | Cost, Copyright and Other Restrictions, Description |
| Relation | Beziehungen zwischen den Ressourcen | Kind, Resource (Identifier, Description, CatalogEntry) |
| Annotation | Bemerkungen bez. der Ressource | Person, Date, Description |
| Classification | Position einer Ressource im Klassifikationssystem | Purpose, TaxonPath (Source, Taxon Id, Taxon Entry, Description, Keywords) |

Die LOM-Spezifikation umfasst allerdings nur Basiselemente und -attribute und ist somit jeweils erweiterbar. Andere Spezifikationen von Lernobjekten (IMS, ARIADNE) verwenden Teilbereiche des LOM und ergänzen die Attribute. Die *Metadaten*-Spezifikation der IMS beschreibt ebenfalls Informationen über Lernressourcen [AnWa2000a; AnWa2000b; AnWa2000c]. Diese Spezifikationen basieren auf den Learning Object Metadata der LTSC, wobei die Spezifikationen nach jeweils neuen Entwicklungen der LTSC synchronisiert werden. Es gibt nur geringe Unterscheidungen bezüglich Datentypen oder Katalogen.

Somit ist diese Spezifikation als Basis zu verstehen. Zum Beispiel sollte die Beschreibung der pädagogischen Aspekte eines Lernobjektes wesentlich erweitert werden. Zurzeit ist es weder möglich, die Eignung von Ressourcen für konkrete didaktische Methoden zu bestimmen, noch können pädagogische Planungsdetails (wie zum Beispiel die Kommunikationsstruktur, Evaluation) erschlossen werden. Die Akzeptanz der weniger technisch orientierten Lehrenden und Trainer hängt maßgeblich von derartigen Erweiterungen ab. Für weitere Details sei auf [AdBP2000a] verwiesen.

Ferner müssen für verschiedene Attribute allgemein akzeptierte Ausprägungen (zum Beispiel durch Best Practice) spezifiziert werden: Um eine Lernressource eindeutig einordnen zu können, ist es notwendig, für diese als Ausprägung eine Taxonomie bzw. ein Klassifikationsschema anzugeben. Gerade im universitären Umfeld existieren derartige Taxonomien zur eindeutigen Abbildung von Lerninhalten derzeit nicht. Es wird in der Zukunft notwendig sein, für verschiedene Branchen, Fachbereiche und Domänen Taxonomien zu entwickeln, die eine eindeutige Zuordnung eines Lerninhaltes zulassen.

Abschließend lässt sich feststellen, dass verschiedene Ausprägungen der LOM-Spezifikation bereits in der Praxis verwendet werden. Es ist zu erwarten, dass LOM in den nächsten Jahren an Verbreitung und Akzeptanz gewinnen wird, da mit zunehmender Verbreitung von Lernressourcen über das Internet auch der Bedarf an Methoden und Formaten zur Informationssuche und -verbreitung steigen wird. Eine individuelle Erweiterung ist jedoch dringend erforderlich. Zudem werden Tools und Entwicklungswerkzeuge benötigt, die den Spezifikationsprozess weitgehend automatisieren. Eine integrierte Einbindung in den Entwicklungsprozess von Lernumgebungen und entsprechende Vorgehensmodelle sind für den Erfolg von LOM unerlässlich [Paw12000].

4.4.4 Lernerorientierte Standards

Als Standard wird hier insbesondere eine detaillierte Spezifikation von Daten der Lernenden vorgenommen: *Public and Private Information for Learners (PAPI)*. [LTSC2000b]. Weiterhin werden Standards zur Definition von *Kompetenzen*, *Studentenbezeichner (Student Identifiers)* sowie *Qualität von Selbstlernprozessen (Quality)* erarbeitet. Für diese Standards liegen jedoch bisher nur Anforderungsdefinitionen vor, Spezifikationen werden derzeit erarbeitet und können daher an dieser Stelle noch nicht diskutiert werden.

4.4.4.1 *Public and Private Information (PAPI)*

Der Standard *Public and Private Information for Learners (PAPI)* gliedert Informationen über den Lernenden in sechs Kategorien [IEEE2000b]. Dabei sollen persönliche Informationen, Präferenzen, Leistungen, Sicherheitsinformationen, Beziehungen und Kompetenzen (Portfolio) eines Lernenden dauerhaft und austauschbar dargestellt und für Lernumgebungen verfügbar gemacht werden. Die Kategorien von PAPI sind in Abbildung 39 dargestellt.

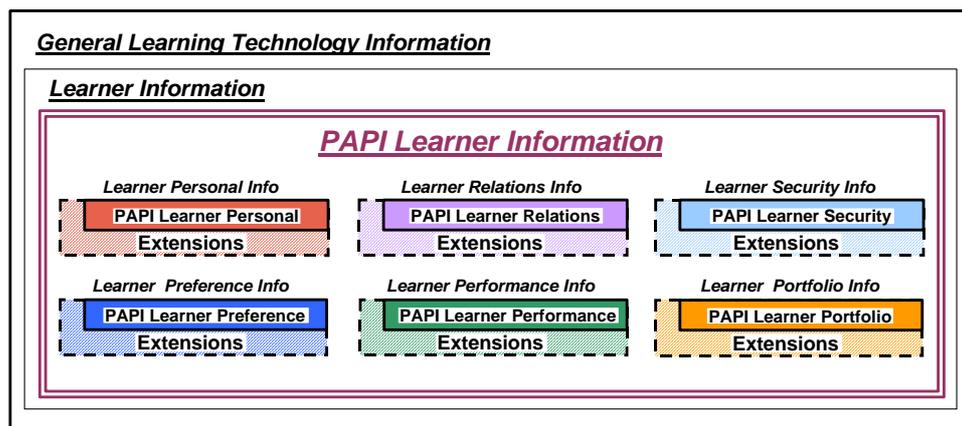


Abbildung 39: PAPI Learner Information [IEEE2000c]

Die Kategorien, Elemente und deren Beschreibungen sind in Tabelle 20 zusammengefasst:

Tabelle 20: Kategorien der Public and Private Learner Information

| Typ | Beschreibung | Beispielattribute |
|--------------------|--|--|
| Personal | allgemeine private Informationen, die nicht direkt in Verbindung mit der Leistung des Lernalers stehen | Name, Adresse, Alter |
| Preference | Benutzerpräferenzen und Merkmale | Inhalt, Lernstil, Lernort, verfügbare Technologie, physische Behinderungen |
| Performance | Vergangenheits-, Gegenwarts- und Zukunftsinformationen über die Lernleistung des Benutzers | Zeitstempel, Leistungskodierung, Zertifizierung |
| Portfolio | Informationen über Leistungen und Arbeiten des Benutzers | Kurse, Veröffentlichungen |
| Relations | Beziehungen zu Akteuren während der Lernprozesse | Lehrende, Kommilitonen, Gruppen, Art der Beziehung |
| Security | Informationen zu Sicherheitsaspekten | Passwörter, private keys, public keys |

Es ist ersichtlich, dass ein solcher Standard, insbesondere für adaptive, intelligente Lernumgebungen oder Administrationssysteme, Informationen zur Verfügung stellt, die zur Individualisierung verwendet werden können. Dennoch sind die Elemente für ein umfassendes Lernermodell nicht ausreichend, so werden zum Beispiel keine motivationalen Aspekte erfasst. Aufgrund dieser Schwierigkeiten ist die Verabschiedung und eine entsprechende Nutzung dieses Standards derzeit nicht zu erwarten. Weiterhin ist momentan nicht geklärt, welche Daten global verwendet werden können, ohne gegen landesspezifische Datenschutzbestimmungen und ethische Grundsätze zu verstoßen. Die Heterogenität recht-

licher Aspekte im europäischen Raum und den dort implementierten Ausbildungssystemen steht der Nutzung des vollständigen PAPI-Standards entgegen. Spezifisch angepasste Ausschnitte der PAPI können jedoch schon genutzt werden. Letztendlich muss ein transparenter Standard entwickelt werden, der die Privatsphäre der Lernenden schützt, aber die Individualisierung von Lernumgebungen fördert. Dabei muss der Lernende jederzeit die Verwendung persönlicher Daten kontrollieren können und die Verwendung autorisieren.

4.4.4.2 *Learner Information Package (LIP)*

Der mit PAPI vergleichbare Standard *Learner Information Package (LIP)* wurde im IMS-Projekt entwickelt [SmTS2001]. Er übernimmt und reorganisiert die Kategorien aus PAPI und bietet ebenfalls Informationen, die zur Adaptation verwendet werden können. Das Modell ist als Basismodell zu verstehen, das erweitert werden muss, um ein individuelles Lernermodell erzeugen zu können. Die folgende Tabelle führt die einzelnen Kategorien des LIP im Vergleich zu denen des PAPI auf.

Tabelle 21: Gegenüberstellung PAPI - LIP

| PAPI | LIP |
|-------------|--|
| Personal | Identification |
| Preference | Accessibility |
| Performance | Transcript, Goal, Qualifications, Activity |
| Portfolio | Activity, Interest, Competency |
| Relations | Affiliation, Relationship |
| Security | Security Key |

4.4.4.3 *Reusable Competencies Definitions*

Der Standard *Reusable Competencies Definitions* der IMS beschreibt die Definition von Kompetenzen Lernender [AnKe2001]. Kompetenzen können in Lernumgebungen genutzt werden, um Vorwissen abzubilden. Ferner bietet dieser Standard eine Schnittstelle zu externen Applikationen im Bereich der Personalentwicklung und ein Modell zur Repräsentation von Kompetenzen. Die komplexe Beschreibung der Einordnung und möglichen Nutzung einer Kompetenz wird in einem Modell (entsprechend den Katalogen der IEEE Standards) abgebildet. Dieses kann jedoch noch nicht standardisiert werden, da derzeit nur organisationsspezifische Modelle verwendet werden. Die Abbildung der Kompetenzen und der Nutzen des Austausches sind daher von der Qualität dieses Modells abhängig.

4.4.4.4 *Universal Learning Format*

Das Universal Learning Format bezieht sich nicht ausschließlich auf lernorientierte Spezifikationen, sondern integriert verschiedene Spezifikationen. Der Fokus liegt auf der Betrachtung lernerspezifischer Informationen [Saba2000]:

Learning Content Format: Diese Spezifikation definiert eine hierarchische Kursstruktur, Lerninhalte und Überprüfungen. Diese basieren auf den Standards CSF, SCORM (siehe 4.5), Content Packaging (siehe 4.5.1) und QTI (siehe 4.4.6.2) und werden daher hier nicht erneut erläutert.

Das *Catalog Format* beschreibt ein Format zur Spezifikation von Katalogen für Lernumgebungen. Dabei werden Ressourcen durch die Beschreibung von Schlagworten, Rechten, Kosten, Kompetenzen zur Recherche und zum Ankauf verfügbar gemacht.

Das *Competency Format* beschreibt ein Netzwerk von Kompetenzen, die durch Lernprozesse erworben wurden. Dadurch können Kataloge adäquat spezifiziert und durchsucht werden. Eine wichtige Funktion zur Interoperabilität sind Abbildungsfunktionen zu anderen Kompetenzdefinitionen.

Das *Certification Format* ermöglicht eine interoperable Darstellung von Zertifizierungen, die zwischen Organisationen ausgetauscht werden können. Zudem kann eine Anpassung von Lerninhalten auf Basis bestehender Zertifizierungen erfolgen.

Das *Profile Format* ist vergleichbar mit den bisher beschriebenen Lernalterspezifikationen. Dabei werden persönliche Daten, Zielsetzungen, Lernhistorie, Kompetenzen und Zertifizierungen abgebildet; somit werden die oben beschriebenen Spezifikationen für einen Lernenden instanziiert. Auch hier sind das Autorisierungsproblem und Aspekte des Datenschutzes nicht geklärt.

Der Standard integriert zwar verschiedene andere Standards, dennoch sind die Probleme der Abbildung von Lernerprofilen nicht gelöst. Zudem muss eine Internationalisierung (z. B. bezüglich der Kompetenzen oder Zertifizierungen) erfolgen, um diesen Standard vergleichbar und sinnvoll einsetzbar zu machen.

4.4.5 Inhaltsorientierte Standards

Inhaltsorientierte Standards beziehen sich auf die Rekombination, Reorganisation und Rekontextualisierung von Lerninhalten, um Lerninhalte system- und anwendungsunabhängig nutzen zu können. Diese Standards sollen im wesentlichen Entwickler bei einer effizienten Erstellung computerunterstützter Lernumgebungen unterstützen. Als wesentliche Standards in der Entwicklung sind *Course Sequencing* und *Content Packaging* zu nennen. Diese Standards spezifizieren Methoden, die das Zusammenstellen und den Ablauf von verschiedenen Inhalten unterstützen.

4.4.5.1 Content Packaging

Im IMS-Projekt wurde eine *Content Packaging*-Spezifikation entwickelt [AnKe2001b; AnKe2001c; AnKe2001d]. Insbesondere Hersteller von Lernsoftware und Lernumgebungen sollen unter Verwendung dieser Spezifikationen die Möglichkeit erhalten, größere Zielgruppen zu erreichen. Zudem soll die Erstellung von Lernumgebungen vereinfacht und beschleunigt werden. Die Basisstruktur ist in Abbildung 40 dargestellt.

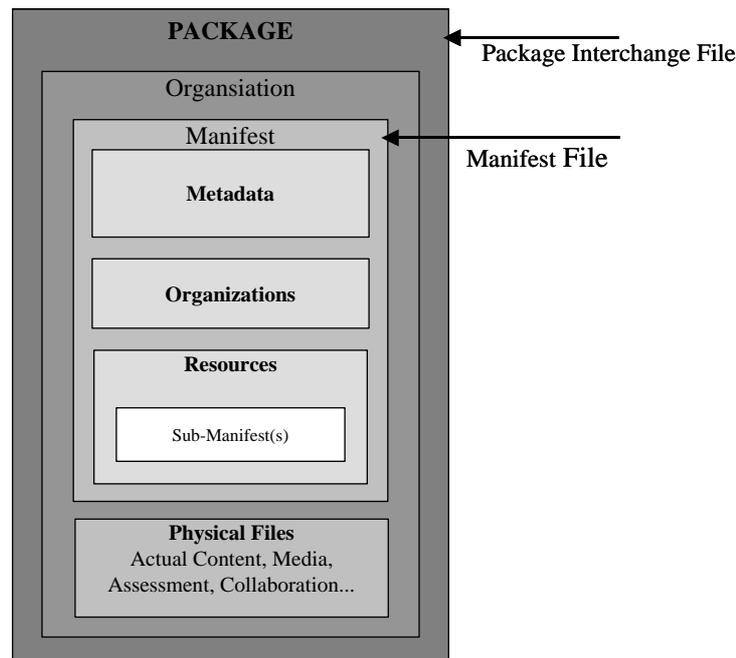


Abbildung 40: IMS Content Packaging

Zentraler Aspekt dieses Standards ist die Beschreibung von Packages, die als wiederverwendbare Einheiten gesehen werden und als solche distribuiert werden können. Diese bestehen aus den tatsächlichen Dateien der Lernumgebungen und einem Manifest, welches Metadaten, die Struktur (Organisation) der einzelnen Einheiten und den Verweis auf die Ressourcen enthält.

In Kooperation mit ADLNET wurde dieser Standard in SCORM (Version 1.2) integriert, was zu einem hohen Verbreitungsgrad führen wird. Eine detaillierte Beschreibung zum SCORM erfolgt im eigenständigen Abschnitt (siehe 4.5).

4.4.5.2 Learning Material Markup Language

Die *LMML (Learning Material Markup Language)* ist eine Auszeichnungssprache zur Strukturierung von einzelnen Lerninhalten bzw. Lernmaterialien. LMML wurde im Rahmen des Forschungsprojekts PaKMaS (Passauer Knowledge Management System) an der Universität Passau entwickelt [PaKM2000]. LMML ist eine XML-Applikation zur Repräsentation des Inhalts und der Struktur von Lernmaterialien. Lernmaterialien werden in LMML durch Sektionen strukturiert und um Inhaltsobjekte ergänzt, die sich in Strukturobjekte (Listen, Tabellen) und Medienobjekte (Text, Sound, Image, Animation) gliedern. Einem Inhaltsmodul wird jeweils eine semantische Charakterisierung (Motivation, Definition, Paragraph, Illustration, Beispiel, Übung, Bemerkung) zugeordnet. LMML wurde bereits um semantische Auszeichnungen für eine Anwendung in der Informatik erweitert. Durch die Verwendung dieser Objekte können einfache Lerneinheiten übersichtlich ge-

gliedert werden. [SuFr2001] betonen, dass die Verwendung unterschiedlicher Unterrichtsmethoden unterstützt wird. Die Durchführung verschiedener Unterrichtsformen wird implizit modelliert, d. h., es werden unterschiedliche Darstellungsformen verwendet.

Diese Art der Auszeichnung eignet sich für die Präsentation von Inhalten und gibt eine intuitive Übersicht über die semantische Struktur eines Dokuments, dennoch lassen sich keine umfangreichen didaktischen Methoden modellieren und komplexe Benutzeranpassungen vornehmen. Daher ist der Wirkungsbereich ohne Erweiterungen durch Metadaten oder andere Standards eingeschränkt.

4.4.6 Managementorientierte Standards

Diese Kategorie enthält Standards, die den Austausch, die Kombination und Administration von Kursen ermöglichen, die in unterschiedlichen Systemen verwendet werden. Dies bezieht sich auf den Austausch bzw. die Portabilität von Kursen und Lerninhalten, von Daten über Lernende sowie über den Datenaustausch in Administrationsprozessen. Als maßgebliche Entwicklung ist hier der Standard *Computer Managed Instruction (CMI)* zu nennen, der den Datenaustausch zwischen Lernmanagementsystemen und Lernobjekten betrachtet. Es ist anzumerken, dass diese Klasse von Standards auch auf die lernerorientierten Standards (siehe 4.4.4) zurückgreift, daher ist eine eindeutige Trennung dieser Standards nicht möglich. Der Standard *Question and Test Interoperability (QTI)* befasst sich mit der Austauschbarkeit von Lernerfolgsüberprüfungen, während der Standard *Enterprise* den Datenaustausch zwischen Lernmanagementsystemen und anderen betrieblichen Informationssystemen betrachtet.

4.4.6.1 *Computer Managed Instruction (CMI)*

Ziel des Standards CMI ist die Interoperabilität von *Lernmanagementsystemen (LMS)*. Kurse unterschiedlicher Formate (wie zum Beispiel CBT-Programme, internetbasierte Lernumgebungen oder Seminare) sollen von zentralen Administrations- und Steuerungssystemen, den LMS, verwaltet werden können [IEEE2000a]. Um Kurse system-, plattform- und applikationsunabhängig integrieren zu können, müssen verschiedene Informationen zwischen dem LMS und den integrierten Lernumgebungen ausgetauscht werden. Weiterhin müssen umfangreiche Daten über den Lernenden und den Lernprozess erfasst und genutzt werden, um die Kursgestaltung individuell anpassen zu können. Der CMI-Standard umfasst somit umfangreiche Informationen zur Integration verschiedener Lernumgebungen mithilfe eines LMS (Tabelle 22). Die technische Umsetzung wird bei der zukünftigen Weiterentwicklung jedoch gegenüber ethischen und rechtlichen Aspekten in den Hintergrund treten. Es ist offensichtlich, dass gerade im Bereich des Datenschutzes und des Schutzes der Privatsphäre der Lernenden kritische Daten erfasst und genutzt werden. Wie bei der PAPI-Spezifikation wird dieser Standard nur dann erfolgreich sein, wenn die Kontrolle persönlicher Daten bei den Lernenden liegt und eine ausreichende Transparenz über die Verwendung der Daten deutlich wird. Durch die Verwendung von CMI in SCORM wird dieser Standard jedoch zunehmende Verbreitung finden.

Tabelle 22: Kategorien der Computer Managed Instruction

| Kategorie | Beschreibung | Beispielattribute |
|-----------------------------|---|--|
| Basisinformationen | allgemeine Informationen, die für jedes System gültig sind | Ort der Lernressourcen, Student ID, Status, Speicher, Dauer, Modus |
| Vergangenheitsdaten | Daten vergangener Sitzungen | spezifische Systeminformationen |
| Aufrufdaten | Daten, die bei jedem Aufruf einer Lerneinheit generiert werden müssen | spezifische Systeminformationen |
| Kommentare | Kommentare des Lehrenden zu einzelnen Lerneinheiten | Anmerkung |
| Evaluation | Informationen über die Leistungsüberprüfung | Zertifikate, Evaluationsform, Bewertungsskala |
| Ziele | Informationen über Leistungen für spezifische Lernziele | Lernziele, Bewertung, Punkte |
| Lernerdaten | Daten über Leistungen und Aktivitäten des Lernenden | Lerneinheiten, Prüfungsleistungen |
| Demographische Daten | Daten des Lernalters vor Kursbeginn | Name, Adresse, Erfahrungen |
| Präferenzen | Benutzerpräferenzen | Sprache, Medienformate, Methoden |

4.4.6.2 Question and Test Interoperability (QTI)

Die *Question and Test Interoperability*-Spezifikation (*QTI*) stellt ein einheitliches Format für die Leistungsüberprüfung von Lernenden zur Verfügung [SmSh2001a; SmSh2001b; SmSh2001c]. Durch die einheitliche Beschreibung von Tests und entsprechenden Ergebnissen können die Lernleistungen und Lernerfolge von verschiedenen Systemen verwendet werden. Gerade im Bereich adaptiver, intelligenter Systeme wird somit eine vereinfachte Erstellung eines Benutzerprofils ermöglicht, Prüfungen werden nicht redundant durchgeführt. Tabelle 23 zeigt eine Übersicht über die Objekte dieser Spezifikation.

Dieses Spezifikationsformat ermöglicht den Austausch von Informationen zwischen einzelnen Prüfungssystemen, zum Lernmanagementsystem und zu den entsprechenden Lernerprofilen. Durch die formale Abbildung der Prüfungsstruktur werden die Bewertungen nachvollziehbar. Dennoch sind mithilfe dieses Formates nur einfache Prüfungsmethoden möglich. Sobald Bewertungen über komplexe Übungsabläufe einfließen, die zudem nicht operationalisierbare Faktoren enthalten, muss das System entsprechend erweitert werden.

Tabelle 23: Objekte des QTI

| Objekt | Beschreibung |
|-----------------------------|---|
| Assessment | Datenstruktur einer Prüfung, bestehend aus Sektionen |
| Section | Teilbereich einer Prüfung, bestehend aus Fragen |
| Item | Objekt einer Frage |
| Activity Selection | Aktivität innerhalb einer Prüfung |
| Accumulation Process | Akkumulation der Einzelergebnisse zu einem Gesamtergebnis |
| Scoring Weights | Gewichtungen der Bewertung |
| Response Processing | Verarbeitung der Benutzereingaben |
| Presentation | Präsentation der Inhalte und Rückmeldungen |
| Examinee Record | Ergebnisse des Geprüften |
| Outcomes | zu bewertende Ergebnisse |
| Response | Antworten des Benutzers |
| Flow | Präsentationsstruktur, bestehend aus einzelnen Blöcken |
| Material | Inhalte, die den Benutzern präsentiert werden |

4.4.6.3 Enterprise

Die *Enterprise*-Spezifikation des IMS-Projekts beschreibt den Datenaustausch zwischen Lernumgebungen/Lernmanagementsystemen und anderen Applikationen in Unternehmen [AnCV1999; CoVe1999a; CoVe1999b]. Dabei werden Daten zum Beispiel zwischen personalwirtschaftlichen Applikationen, digitalen Bibliotheken oder Administrationssystemen ausgetauscht. Mithilfe eines solchen Datenaustausches sollen Inkonsistenzen und Medienbrüche vermieden werden.

4.4.7 Didaktische Standards

Der Begriff der didaktischen Standards kann bezüglich der Ebenen des didaktischen Handelns missverständlich interpretiert werden. In diesem Zusammenhang soll die Darstellung didaktischer Komponenten der D- und E-Ebene, die die Entwicklung und Implementierung von Anwendungssystemen betreffen, vorgenommen werden. Dabei steht die Darstellung didaktischer Methoden im Vordergrund. Es hat sich gezeigt, dass die derzeitige Spezifikation von Metadaten durch LOM keine adäquate Repräsentation von didaktischen Inhalten zulässt [Kope2001]. Zudem kann durch diese Beschreibung keine angemessene Zuordnung von Lernobjekten zu didaktischen Kontexten erfolgen. Diese Schwachpunkte sollen durch didaktische Standards behoben werden. Zwar wurde bereits eine Vielzahl an Ansätzen für spezifische didaktische Kontexte erstellt, dennoch sind diese Ansätze bisher nicht in einen Kontext mit anderen Standards gestellt worden. Die folgenden Modelle und korrespondierenden Auszeichnungssprachen decken Teilbereiche der didaktischen Modellierung ab:

Die *Educational Modeling Language (EML)* basiert auf einem Metamodell zur pädagogischen Modellierung von Lernumgebungen. Zentraler Aspekt ist die Einbettung von Lern-

objekten in einen didaktischen Kontext [Kope2001]. Aufgrund der umfangreichen Spezifikation, die sich nicht auf didaktische Modellierung beschränkt, wird dieses Modell in Abschnitt 4.6 gesondert behandelt.

Die *Tutorial Modelling Language (TML)* [Netq2000] ist eine Auszeichnungssprache zur Entwicklung von Tutoriellen Systemen und bezieht sich somit explizit auf einen didaktischen Ansatz. Mit TML lassen sich Frageszenarios entwickeln, die durch die Spezifikation von Fragen, Antworten, Regeln und Hilfestellungen umgesetzt werden. Zwar lassen sich so einfache behavioristische Systeme entwickeln; diese gehen jedoch nicht wesentlich über die Mächtigkeit von QTI hinaus. Zur Entwicklung umfangreicher intelligenter tutorieller Systeme ist diese Auszeichnungssprache nicht geeignet.

Die *Instructional Material Description Language (IMDL)* [Gaed2000] bildet die Struktur, Inhalt, Überprüfungen, Metadaten und ein Lernerprofil ab. Der Ansatz ist geht strikt nach den Prinzipien des Instructional Design vor und impliziert somit eine Festlegung auf bestimmte didaktische Vorgehensweisen. Daher ist auch diese Auszeichnungssprache kein umfassender Ansatz zur gleichberechtigten didaktischen wie technologischen Modellierung.

Es zeigt sich, dass zwar verschiedene Spezifikationen existieren, die Ansätze zu bestimmten didaktischen Methoden liefern. Dennoch existiert derzeit kein Ansatz zur konsistenten und flexiblen Modellierung didaktischer Methoden.

4.5 Sharable Content Object Reference Model

In diesem Abschnitt wird das *Sharable Content Object Reference Model (SCORM)* [Dodd2001b] als Beispiel für einen Standard vorgestellt, der verschiedene Lerntechnologiestandards integriert. Aufgrund der Beteiligung der maßgeblichen Standardisierungsinitiativen (LTSC, IMS, ARIADNE, AICC) ist dieser Standard als besonders erfolgversprechend anzusehen. Weitere Integrationsansätze werden nachfolgend angesprochen.

SCORM ist ein Referenzmodell zur Integration verschiedener Standards. Die Zielsetzung von SCORM ist es, ein Referenzmodell für webbasierte Lernmanagementsysteme zur Verfügung zu stellen, die system- und plattformunabhängig Lerneinheiten verwenden und verarbeiten können. Es besteht aus zwei maßgeblichen Komponenten:

- Das *Content Aggregation Model* (Inhaltsaggregationsmodell) ist eine Repräsentationsform zur Zusammenstellung von Lernsequenzen aus einzelnen Lernobjekten. So soll die Möglichkeit geschaffen werden, einzelne Lerneinheiten in organisationsübergreifenden Repositories abzulegen und daraus neue Inhalte und Module zusammenzustellen.
- Die *Run-Time Environment* (Laufzeitumgebung) stellt eine Schnittstelle zwischen Lernmanagementsystem und einzelnen Lerneinheiten zur Verfügung. Dabei sollen Lernobjekte unabhängig von einer LMS-Instanz genutzt werden können.

Die beiden zentralen Elemente aus SCORM werden im Folgenden im Detail beschrieben und nachfolgend diskutiert. Dabei wird die Version SCORM 1.1 [Dodd2001a] verwendet,

da diese derzeit in ELM implementiert wurde. Die Änderungen zur Nachfolgeversion SCORM 1.2 [Dodd2001b, Dodd2001c, Dodd2001d] werden dabei jeweils angesprochen.

4.5.1 Content Aggregation Model

Das Content Aggregation Model beschreibt die Aggregation einzelner Lernobjekte zu einer Kursstruktur. Dabei sind folgende Elemente Teil des Modells:

Das *Content Structure Format (CSF)* ist ein Format zur Definition von Kursobjekten, das die Struktur, Metadaten und Referenzen zu externen Objekten zusammenfasst. Die kleinste in Lernprozessen eigenständig verwendbare Einheit sind dabei *Sharable Content Objects (SCO)*. CSF setzt sich zusammen aus Kurs-Metadaten (Course Metadata) und der Verknüpfung zu Blöcken (Blocks) sowie Sharable Content Objects (SCO) zusammen. Dabei sind Blöcke als Aggregation von SCOs und weiteren Blöcken definiert, sie sind also als zusammengesetzte Lerneinheiten anzusehen. Durch diese hierarchische Gliederung wird ein Kurs bezüglich seiner Struktur, den Lerneinheiten und Metadaten vollständig beschrieben. Ein zu SCORM kompatibles LMS kann somit Kurse in Form des CSF importieren und exportieren. Es ist allerdings anzumerken, dass CSF in der Version 1.1 nur auf SCOs und Metadaten referenziert und diese sich nicht physisch transportieren lassen, so dass ein ganzer Kurs in einem festgeschriebenen Format distribuiert werden kann. Diese Erweiterung ist in der Version 1.2 durch die Verwendung des Content Packaging Standards erfolgt und ist in Abbildung 41 bereits berücksichtigt [vgl. Dodd2001d].

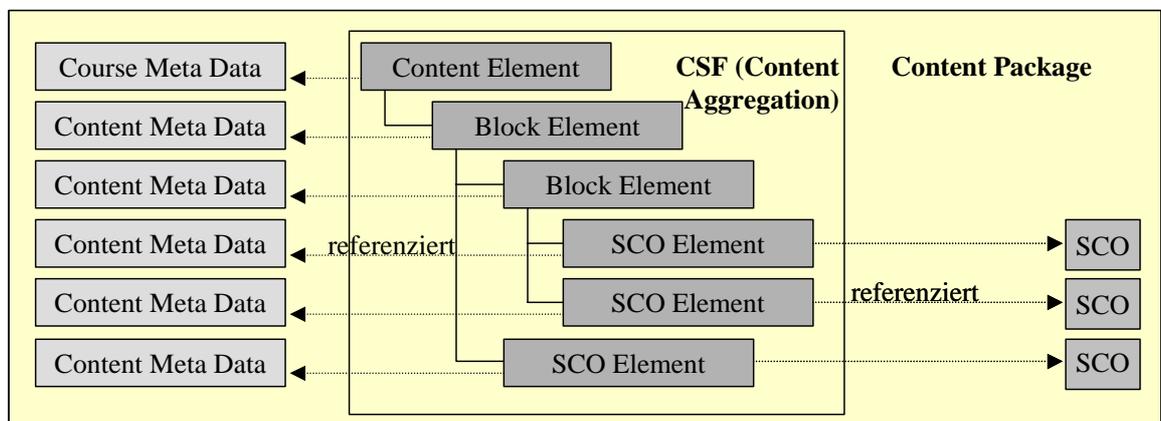


Abbildung 41: Inhaltsaggregation in SCORM [Dodd2001a]

Zentraler Bestandteil dieses Modells sind SCOs, die auch als Lerneinheiten angesehen werden können und die Lerninhalte enthalten. SCOs setzen sich aus einzelnen Assets (Medienobjekten) zusammen, wie zum Beispiel Audio, Video oder Text. Zudem muss ein SCO ein Application Programming Interface (API) enthalten, das für die Kommunikation mit dem LMS verantwortlich ist und zum Beispiel den Aufruf eines SCOs steuert (siehe 4.5.2).

Metadaten basieren in SCORM auf der IMS-Metadatenspezifikation; sie werden auf verschiedenen Ebenen spezifiziert:

- *Course Metadata* enthalten umfangreiche Metadaten zur Aggregatsebene, die durch CSF repräsentiert ist. Dadurch ist die Möglichkeit gegeben, diese Aggregationsebene z. B. in Kursdatenbanken zu durchsuchen und zu nutzen.
- *Content Metadata* sind Metadaten zu Blöcken und SCOs. Dadurch können Repositories nach einzelnen Lerninhalten durchsucht, rekombiniert und wiederverwendet werden.
- *Raw Media Metadata* sind Metadaten zu einzelnen Assets. Da Assets nicht alleinstehend in einem Lernprozess eingesetzt werden können, unterscheiden sich diese Metadaten zu Course und Content Metadata dadurch, dass mehr Elemente optional spezifiziert werden können, d. h., sie sind keine Pflichtelemente.

Die Struktur des CSF ist in Tabelle 24 beschrieben:

Tabelle 24: Content Structure Format

| Kategorie | Beschreibung | Elemente |
|--------------------------|---|---|
| Content | Container-Element zur Zusammenfassung der Subelemente | Global Properties, Block |
| Global Properties | Globale Eigenschaften eines Kurses | externe Metadaten (Quelle, Modell), Curriculare Taxonomie (Modell) |
| Block | Beschreibung der Komponenten, die im CSF enthalten sind | Bezeichner externe Metadaten Identifikation (Titel, Beschreibung, Label) Vorbedingungen (Typ) Block (Subblock) SCO Block Alias |
| SCO | Beschreibung eines SCOs (Subelement eines Blocks) | Bezeichner externe Metadaten Identifikation (Titel, Beschreibung, Label) Vorbedingungen (Typ) zeitliche Beschränkung (maximale Zeit, Aktion bei Überschreitung) Aufruf (Ort, Parameter-String, Punktzahl zur Bewältigung, Alias) |

4.5.2 Run-Time Environment

Die Laufzeitumgebung von SCORM ermöglicht die Datenübergabe und den Aufruf web-basierter Inhalte durch ein beliebiges SCORM-konformes LMS. Dabei wird ein SCO durch die Funktion *Launch* via HTTP gestartet. Dieser Mechanismus ermöglicht die Kommunikation zwischen LMS und SCO durch die Benutzung einer gemeinsam genutzten *API*. Die API enthält jeweils den Zustand eines SCO, zudem werden Daten zwischen LMS und SCO versendet. Zu Beginn eines Lernprozesses muss das LMS die nachfolgenden Funktionen zur Verfügung stellen. Es ist Aufgabe der SCO-API, die entsprechende API des LMS aufzufinden. Dieses wird durch die Vorschrift erleichtert, dass eine Session jeweils in einem zusätzlichen Fenster des LMS aufgerufen wird. Die Implementierung kann seitens des LMS z. B. über Servlets, Active Server Pages (ASP) oder Common Gateway Interface (CGI) erfolgen. Eine Client-seitige Implementierung kann in JavaScript oder über Java-Applets erfolgen, die Implementierung ist nicht genauer reglementiert. Es müssen lediglich folgende standardisierte Funktionen beherrscht werden:

- *LMSInitialize(“”)* und *LMSFinish(“”)* regeln den Ausführungszustand.
- *LMSGetLastError()*, *LMSGetErrorString(errornumber)* und *LMSGetDiagnostic(parameter)* behandeln das Auftreten von Fehlern. Dabei sind verschiedene Standardfehler in Form einer Diagnose (z. B. Verbindungsunterbrechung) vordefiniert.
- *LMSGetValue(data model element)*, *LMSSetValue(data model element, value)* und *LMSCommit(“”)* regeln den Datenaustausch zwischen LMS und SCO. Dabei können nur Werte ausgetauscht werden, die im Datenmodell enthalten sind.

Ein *Datenmodell* spezifiziert die Daten, die zwischen LMS und SCO ausgetauscht werden. Als Datenmodell wird dabei der Standard CMI verwendet (siehe 4.4.6.1). Es müssen mindestens die Elemente bezüglich Studentenidentifikation, Lerneinheit, Punktschema, Aufruf- und Abbruchdaten übergeben werden, um eine eindeutige Identifizierung eines Lernprozesses zu ermöglichen. Abschließend zeigt Abbildung 42 eine schematische Darstellung der Laufzeitumgebung.

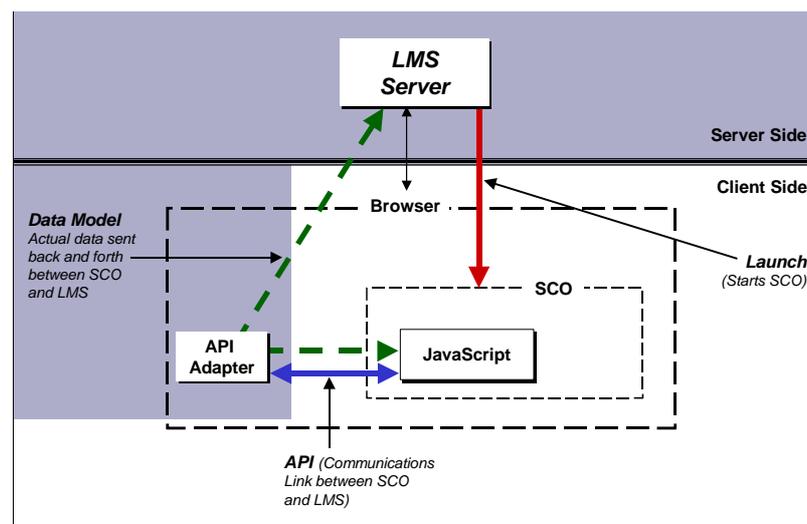


Abbildung 42: SCORM Run-Time Environment [Dodd2001a]

4.5.3 Kritische Würdigung

Die Beschreibung durch CSF ermöglicht die Identifikation, die Definition und die Wiederverwendung von Lernobjekten verschiedener Ebenen. Dabei werden Strukturen, Metadaten und Laufzeitverhalten in standardisierter Form zur Verfügung gestellt. Das Modell bietet so eine einsatzfähige Umsetzung unter Beachtung von Metadaten sowie managementorientierten und inhaltsorientierten Standards. Die Möglichkeiten von Entwicklern werden durch SCORM nicht eingeschränkt. Dennoch ist die jetzige Version noch nicht ausgereift anzusehen: So sind derzeit nur geringe Möglichkeiten zur Adaptation an Lernende möglich. Daher müssen die Datenmodelle, die zum Austausch übergeben werden, in verschiedene Richtungen erweitert werden (Lernermodelle, Methodenmodelle).

Ein weiterer kritischer Akzeptanzfaktor ist die Umsetzung und Verwendung von Lerninhalten, die nicht SCORM-konform sind. Bereits eine zusätzliche Spezifikation von Metadaten ist für viele Entwickler ein Grund, Lerntechnologiestandards nicht in ihre Entwicklungen einzubeziehen. Daher müssen Abbildungsverfahren für verschiedene Lernumgebungstypen geboten werden, die eine einfache Konvertierung zulassen.

Da SCORM derzeit das einzige Modell ist, das eine durchgehende, implementierbare Lösung zur Integration der Standards anbietet, wird es im Essener-Lern-Modell verwendet.

4.6 Educational Modelling Language

Es hat sich gezeigt, dass die derzeitige Spezifikation von Metadaten durch LOM keine adäquate Repräsentation von didaktischen Inhalten zulässt und daher erweitert werden sollte [vgl. Kope2001]. Das SCORM-Konzept betrachtet die reine Zusammenfassung von Lerninhalten zu Packages, ohne dabei auf didaktische Kontexte oder die korrespondierende Eignung der Lerninhalte einzugehen. Weiterhin wurde in 2.3.8 gezeigt, dass derzeit keine standardisierte Repräsentation didaktischer Methoden existiert.

Die *Educational Modelling Language (EML)* basiert auf einem Metamodell zur Modellierung pädagogischer Aspekte von Lernumgebungen. Zentraler Aspekt ist die Einbettung von Lernobjekten in einen didaktischen Kontext. [Kope2001] beschreibt vier Komponenten des Metamodells:

Die *Theories of learning and instruction (Lern- und Lehrtheorien)* beschreiben Theorien, Prinzipien und Modelle des Lernens und Lehrens. In diesem Modell werden die Theorien als empiricist (behavioristisch), rationalist (kognitivistisch und konstruktivistisch), pragmatist-sociohistoric (sozio-konstruktivistisch) und eclectic (Mischformen) klassifiziert.

Das *Learning Model (Lernermodell)* beschreibt, wie Lernende konsensbasiert lernen. Das Lernmodell besteht aus der Beschreibung von Interaktionen in bestimmten Lernsituationen.

Das *Domain Model (Domänenmodell)* ist eine Abbildung der Anwendungsdomäne, für die Lernprozesse initiiert werden.

Das *Units of Study Model (Modell der Lerneinheiten)* bildet ab, wie Lerneinheiten bei gegebenen Lerntheorien, Lernermodellen und Domänenmodellen gestaltet werden können.

Tabelle 25 fasst die wesentlichen Beschreibungselemente und Zusammenhänge einer Lerneinheit zusammen:

Tabelle 25: Informationsmodell EML

| Element | Beschreibung |
|---------------------|---|
| Unit-of-study | Basiseinheit der Spezifikation in unterschiedlicher Granularität (Curriculum, Kurs, Lerneinheit) |
| Metadata | Metadaten |
| Roles | Rollen der Akteure innerhalb eines Lernprozesses. Dabei wird zwischen Learner und Staff unterschieden, die jeweils verschiedene Rollen annehmen können. |
| Learning-objectives | Lernziele bzw. gewünschter Output eines Lernprozesses |
| Prerequisites | Vorbedingungen einer Lerneinheit (Persönlichkeitsmerkmale, Leistungen) |
| Content | Lerninhalte einer Lerneinheit, beschrieben durch Activities und Environment |
| Activity | Aktivitäten beschreiben einzelne Schritte in einem Lernprozess (z. B. Erarbeitung, gemeinsame Ausarbeitung). |
| Environment | Beschreibt eine Lernsituation durch verschiedene Objekte: Kommunikationsobjekte, Wissensobjekte, Werkzeugobjekte, Fragebogenobjekte, Ankündigungsobjekte, Index-/Suchobjekte |
| Method | Methode beschreibt den Ablauf eines Lernprozesses durch die Beschreibung der Aktivitätsstrukturen (Abfolge, Bedingungen), Rollenbeziehungen und frei definierbaren Bedingungen. |

Neben diesem Referenzmodell ist EML als XML-Applikation verfügbar und bietet somit ebenfalls Werkzeuge zur Implementierung an¹. Der Ansatz von EML unterstützt die in Abschnitte 3.4 und 5.1 geforderten Anforderungen an eine ausgewogene didaktische und technologische Modellierung von Lernumgebungen. EML ist jedoch nicht in ein Vorgehensmodell eingebunden, daher müssen zur sinnvollen Einsetzbarkeit weitere Unterstützungsfunktionen verfügbar gemacht und die Anpassung an spezifische Projektsituationen ermöglicht werden. Des Weiteren werden genaue Kontextanalysen, Querschnittsfunktionen wie das Projektmanagement sowie Qualitätssicherung im Modell derzeit noch nicht erfasst.

¹ Das Fachgebiet Wirtschaftsinformatik der Produktionsunternehmen ist Kooperationspartner in einem Projekt zur Weiterentwicklung der EML, da die Themenbereiche von EML und des Essener-Lern-Modells eng miteinander verknüpft sind. Daher werden beidseitig Ansätze zur Weiterentwicklung eines umfassenden Modellierungsansatzes in die jeweiligen Forschungsarbeiten einfließen.

4.7 Folgerungen

In den vorhergehenden Abschnitten wurde beurteilt, welche der beschriebenen Standards den Anforderungen aus Abschnitt 4.1 genügen. Im Folgenden werden abschließend die Erfolgsaussichten bez. der Akzeptanz und der potenziellen Nutzung betrachtet.

Zur Beschreibung von Lernressourcen sind insbesondere Learning Object Metadata weit verbreitet [vgl. Duva2001, HNWW2001, KDKS2001]. Zudem wird der LOM-Standard derzeit in der ISO/IEC als Vorlage für eine Norm zur Beschreibung von Lernressourcen diskutiert, was die weitere Verbreitung fördern kann. Aus inhaltlicher Sicht deckt LOM die Anforderungen an eine umfassende Beschreibung noch nicht vollständig ab: Insbesondere der didaktische Kontext eines Lernobjekts ist noch unzureichend modelliert. LOM sollte somit als Basis verwendet werden und muss bez. didaktischer Aspekte erweitert werden.

Im Bereich der Lernerbeschreibungen sind derzeitige Spezifikationen unzureichend. Zwar werden umfassende Daten über Lernende erfasst, dennoch genügen diese Daten nicht zu einer individuellen Anpassung von Lernumgebungen. Aufgrund der Datensicherheit und dem Schutz der Privatsphäre der Lernenden sind zudem Akzeptanzprobleme zu erwarten.

SCORM bietet einen Standard zur Interoperabilität von Lernmanagementsystemen und Lernumgebungen. Die Version 1.2 definiert ein austauschbares Format für Lernumgebungen und Kurse durch die Integration der Standards LOM, CMI und Content Packaging. Aufgrund dieser Integration und der Verbreitung in verschiedenen LMS ist dieser Standard als erfolgversprechend zu beurteilen. Erweiterungsbedarf besteht in den Bereichen der Adaptation und der Modellierung didaktischer Methoden.

Im Bereich der Modellierung didaktischer Methoden ist die Educational Modelling Language erfolgversprechend, da sie ein umfangreiches Modell zur Beschreibung didaktischer Szenarien zur Verfügung stellt. Dennoch fehlen in diesem Modell Aktivitätsbereiche der Qualitätssicherung und der Kontextanalyse.

Abschließend lässt sich feststellen, dass insbesondere didaktische Aspekte in den maßgeblichen Standardisierungsaktivitäten der IEEE LTSC, der IMS-Projekts und SCORM vernachlässigt wurden. Daher besteht der Bedarf, ein Modell zur Beschreibung didaktischer Szenarien zu entwickeln, das bestehende Standards einbezieht. So müssen Schnittstellen zu LOM und SCORM bereitgestellt werden, um ein integriertes Modell und keinen isolierten Standard zu liefern.

Des Weiteren müssen Aspekte der Qualitätssicherung zunehmend einbezogen werden. So liefert kein Standard Richtlinien, die über technische Anforderungen hinausgehen. Somit muss ein integrierter Standard entwickelt werden, der als Richtlinie zur Qualitätssicherung von Lernumgebungen verwendet werden kann und dabei bestehende Beschreibungsstandards einbezieht.

5 Konzeption des Essener-Lern-Modells

In diesem Kapitel wird das Konzept des Essener-Lern-Modells (ELM) beschrieben. Aus der Analyse der didaktischen Grundlagen (Kapitel 2), der Vorgehensmodellierung (Kapitel 3) und der Lerntechnologiestandards (Kapitel 4) ergeben sich klar definierte Anforderungen, die ein Vorgehensmodell zur Entwicklung von Lernumgebungen erfüllen muss. Die Anforderungen und Entwicklungskriterien werden zunächst zusammengefasst. Aufgrund des Umfangs des Modells wird zunächst eine Übersicht über das Essener-Lern-Modell gegeben; im Anschluss werden die Designebenen des Modells im Detail beschrieben.

5.1 Kriterien zur Entwicklung von ELM

Im dritten Kapitel dieser Arbeit wurde gezeigt, dass derzeit existierende Vorgehensmodelle für Lernumgebungen nicht alle Anforderungen der Softwareentwicklung und der didaktischen Planung von Lernumgebungen erfüllen. Wie in Abschnitt 3.2 diskutiert, stehen aus Sicht der Softwareentwicklung insbesondere die Kriterien Benutzbarkeit, Sicherheit, Interoperabilität und Wiederverwendbarkeit im Vordergrund. Besonders die Kriterien der Wiederverwendbarkeit und Interoperabilität von Methoden und Lerninhalten sind in VGM nur unzureichend berücksichtigt. Weiterhin wurde gezeigt, dass existierende VGM häufig nur einzelne Ebenen oder einen partiellen Funktionsumfang einbeziehen und den Entwicklungsprozess daher nicht durchgehend unterstützen. Zur umfassenden Planung von Lernumgebungen müssen die C-, D- und E-Ebenen des didaktischen Handelns in einem Vorgehensmodell betrachtet werden. Dennoch können Erfahrungen der Verwendung verschiedener VGM genutzt werden. Es ist somit notwendig, ein umfassendes Vorgehensmodell zu entwickeln, das einerseits die Stärken bisheriger Modelle nutzt, andererseits die genannten Schwachpunkte behebt. Aus den Beurteilungskriterien für Vorgehensmodelle (siehe 3.2) ergeben sich folgende Anforderungen, die die Basis für das Essener-Lern-Modell darstellen:

- Der *Funktionsumfang (Tätigkeitsbereiche)* muss die umfassende Planung, Konzeption, Entwicklung, Implementierung und Evaluation von Lernumgebungen berücksichtigen. Dafür sind Unterstützungsfunktionen für Management- und Administrationsprozesse, Entwicklungsprozesse, Lernprozesse und QS-Prozesse erforderlich. Es zeigt sich, dass die Verwendung partieller VGM zwar gute Teilergebnisse liefert, der Kontext dabei aber nur unzureichend berücksichtigt wird. Daher müssen Unterstützungsfunktionen für die Curriculumentwicklung und Kontextanalyse (C-Ebene), die Kursentwicklung (D-Ebene) und die Entwicklung einzelner Lerneinheiten (E-Ebene) angeboten werden.
- Die *Integration* von Lernprozessen in einen organisationalen Kontext wird häufig vernachlässigt. So werden Lernprozesse meist von Arbeitsprozessen getrennt betrachtet. Um diese Integration zu ermöglichen, stellt ELM Modelle von Entwicklungs- und Lernprozessen zur Verfügung, die durch Prozessmodelle in Form ereignisgesteuerter Prozessketten (EPK) spezifiziert sind [Sche1998a, Sche1998b, Sche1998c]. Diese umfassen die einzelnen Aktivitäten und Ergebnisse der Prozesse, so dass sie in bestehende Geschäftsprozessmodelle eingebunden werden können. Dies ermöglicht eine verbesserte Planung der Aus- und Weiterbildungsprozesse. Werden in einer Organisation be-

stimmte Kompetenzen benötigt, können Ausbildungsprozesse so gestaltet werden, dass die Kompetenzen zum benötigten Zeitpunkt verfügbar sind (Just-In-Time-Learning).

- Die *Anpassbarkeit* des Modells steht in direkter Verbindung zur Prozessmodellierung. Ein generisches Vorgehensmodell erfordert eine kontextspezifische Anpassung. Die allgemeingültigen Prozesse müssen bezüglich der verschiedenen Kontexte, Aktoren und Einflussfaktoren analysiert und organisationsspezifisch angepasst werden. Als Beispiel seien Prozesse des Projektmanagements genannt: Einerseits können die Unterstützungsfunktionen von ELM, andererseits bestehende Projektmanagementsysteme genutzt werden. Weiterhin können einzelne Teilbereiche aus ELM projektspezifisch angepasst werden. So bietet ELM zum Beispiel die Möglichkeit, verschiedene didaktische Methoden zu implementieren. Es wird dabei keine Methode vorgegeben, die für Entwickler und Lehrende bindend ist; vielmehr können implementierte Methoden erweitert und individuell modifiziert werden.
- *Allgemeine Anforderungen der Softwareentwicklung* müssen im Vorgehensmodell berücksichtigt werden. So sind die Kriterien der Funktionserfüllung, der Zuverlässigkeit und der Wartbarkeit zu beachten. Ermöglicht wird dies durch die Verwendung von evolutionären Prototypen, wodurch die Funktionen von Entwicklern und Benutzern im Entwicklungsprozess stetig angepasst und evaluiert werden. Sämtliche Spezifikationen liegen in einer Datenbank vor und können wiederverwendet und angepasst werden.
- Ein Vorgehensmodell muss interne wie externe Akzeptanzfaktoren umfassen und die *Benutzbarkeit* für alle Aktoren sicherstellen. So müssen Unterstützungsfunktionen allen Aktoren der Entwicklungs-, Administrations- und Lernprozesse verfügbar gemacht werden und an ihre Präferenzen angepasst werden. Zudem ist gerade bei Lernumgebungen die Sicherheit für Entwickler (Urheberrecht, intellektuelles Eigentum) und Lernende (Privatsphäre) zwingend erforderlich.
- *Portabilität/Interoperabilität/Wiederverwendbarkeit*: Lernumgebungen müssen unter verschiedenen Oberflächen, Systemen und insbesondere in verschiedenen Lernmanagementsystemen einsetzbar sein. So erfasst [Häfe2001] mehr als 130 LMS, die unterschiedliche Anforderungen an die Lernumgebungen stellen. Eine Lernumgebung sollte in allen diesen Systemen einsetzbar sein, um eine möglichst weite Verbreitung erreichen zu können. Ein Vorgehensmodell muss also Entwicklungsprozesse derart unterstützen, dass entwickelte Komponenten portabel, interoperabel und wiederverwendbar sind. Aufgrund der hohen Entwicklungskosten multimedialer Lernumgebungen ist dieses Kriterium ein kritischer Erfolgsfaktor für computerunterstützte Lernumgebungen. Dieses wird insbesondere durch die Verwendung von Standards erreicht. Des Weiteren stehen entwickelte Komponenten in einem Repository für weitere Projekte zur Verfügung.
- Ein VGM muss *technologische und didaktische Aspekte* gleichwertig behandeln und ihre Interdependenzen abbilden [vgl. Schu2001]. Dadurch wird die Zusammenarbeit von Aktoren unterschiedlicher Domänen vereinfacht. Zudem gibt das VGM eine Richtlinie, welche Aspekte aus didaktischer und technologischer Sicht beachtet werden sollten, um qualitativ hochwertige Lernumgebungen zu entwickeln. Dabei muss die Entwicklung der Komponenten *Inhalte, Benutzer, Methoden, Präsentation, Kommunikation und Evaluation* unterstützt werden.
- Ein VGM muss für jede Unterstützungsfunktion *Werkzeuge* anbieten, die eine einfache Umsetzung sicherstellen. Dabei müssen kontextabhängige Anforderungen einbezogen

werden. Zu diesem Zweck wurde die ELM-Applikation entwickelt, die alle beschriebenen Prozesse unterstützt (siehe Abschnitt 6.1).

Ein Vorgehensmodell muss diese Anforderungen erfüllen, um eine effiziente und effektive Entwicklung und Nutzung von Lernumgebungen zu ermöglichen. Die konkrete Umsetzung im Essener-Lern-Modell wird in den folgenden Abschnitten detailliert beschrieben.

5.2 Vorgehensstrategie des Essener-Lern-Modells

Das Essener-Lern-Modell basiert auf der Analyse der Vorgehensstrategien und spezifischen Vorgehensmodelle und fasst die Erkenntnisse der Entwicklung von Lernumgebungen zusammen. Dabei wurde in den vorhergehenden Kapiteln gezeigt, dass die Charakteristik des Entwicklungsprozesses verschiedene Anforderungen an das VGM stellt. So unterliegt der Entwicklungsprozess einer ständigen Evaluation und Anpassung: Insbesondere durch ein rechtzeitiges Einbeziehen der Benutzer muss die Lernumgebung unter Umständen auch während der Laufzeit modifiziert werden. Diese Anpassbarkeit wird vor allem durch die Vorgehensstrategie des Spiralmodells (siehe 3.3.6) unterstützt. Aus der zusätzlichen Verwendung einer evolutionären Strategie (siehe 3.3.5) resultieren bereits in frühen Entwicklungsphasen lauffähige Lernumgebungen, die von Lernenden wie Entwicklern evaluiert werden können.

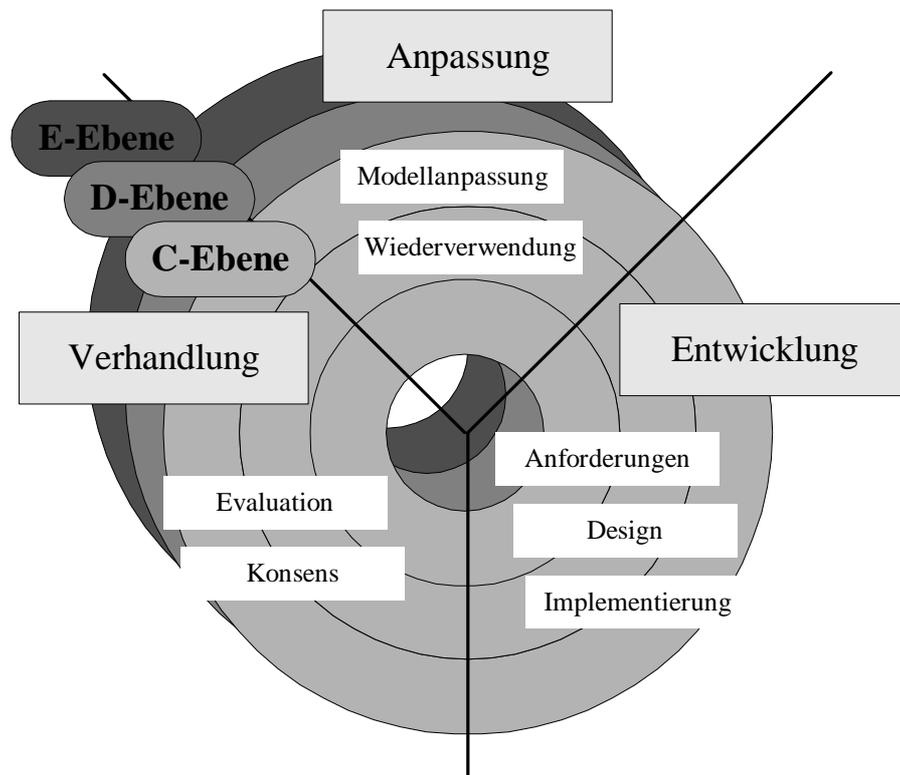


Abbildung 43: ELM-Vorgehensmodell

Das VGM gliedert sich in jeweils drei Entwicklungsschritte und Entwicklungsebenen (Abbildung 43). Die *Entwicklungsschritte* des Modells sind Verhandlung, Anpassung und Entwicklung. Der Schritt der *Verhandlung* umfasst die Planung der jeweils nächsten Aktivitäten und die Evaluation und Abstimmung von Teilergebnissen und entsprechenden Änderungsaufgaben. Diese werden im Projektteam, das Entwickler, Domänenexperten, Manager, Lehrende und Lernende umfasst, abgestimmt. Im Anschluss an die Verhandlung erfolgt die *Anpassung* des Vorgehens. Die Entwicklungsschritte beschreiben also das prinzipielle Vorgehen. Ein konkretes Vorgehen wird jeweils in den Verhandlungsphasen detailliert geplant. Dieses Vorgehen kann projektspezifisch variieren und an die Bedürfnisse einer Organisation angepasst werden. Dazu kommt in dieser Phase eine Analyse und Planung wiederverwendbarer Komponenten. Bereits vor der eigentlichen Entwicklung wird analysiert, welche Komponenten einfließen können. Darauf folgt die *Entwicklung*, die die Aktivitäten der Anforderungsdefinition, Design, Implementierung, Nutzung und Wartung einschließt. Dieser Zyklus wird mehrmals durchlaufen, bis ein Konsens über das Endprodukt besteht.

Die *Entwicklungsschritte* des Essener-Lern-Modells beschreiben die Aktivitäten der Entwicklung von Lernumgebungen. Dabei werden die *Entwicklungsebenen* nach den Ebenen des didaktischen Handelns (siehe 2.3.1.1) bezeichnet:

- Die *C-Ebene (ELM-C)* umfasst die Planung von Ausbildungsmaßnahmen durch eine umfassende Analyse des Kontextes, die Curriculumentwicklung und die Kursorganisation. Querschnittsfunktionen wie Projektmanagement und Qualitätssicherung sind diesem Schritt zugeordnet.
- Die *D-Ebene (ELM-D)* umfasst die Entwicklung von Lernsequenzen (z. B. Kurse, Vorlesungen). Dabei werden Inhalte, didaktische Methoden und die Benutzercharakteristika beschrieben.
- Die *E-Ebene (ELM-E)* betrachtet schließlich die Gestaltung einzelner Lerneinheiten. Dabei bezeichnen Lerneinheiten die kleinsten eigenständig sinnvoll verwendbaren Einheiten. In diesem Schritt werden Kommunikations-, Präsentations- und Evaluationskomponenten gestaltet.

ELM-Run-Time ist eine Zusatzfunktion des Essener-Lern-Modells. Diese Komponente wurde zur besseren Testbarkeit der Prototypen entwickelt. ELM-Run-Time leistet die Basisfunktionalitäten eines Lernmanagementsystems und ermöglicht die Überprüfung der Lernumgebungen.

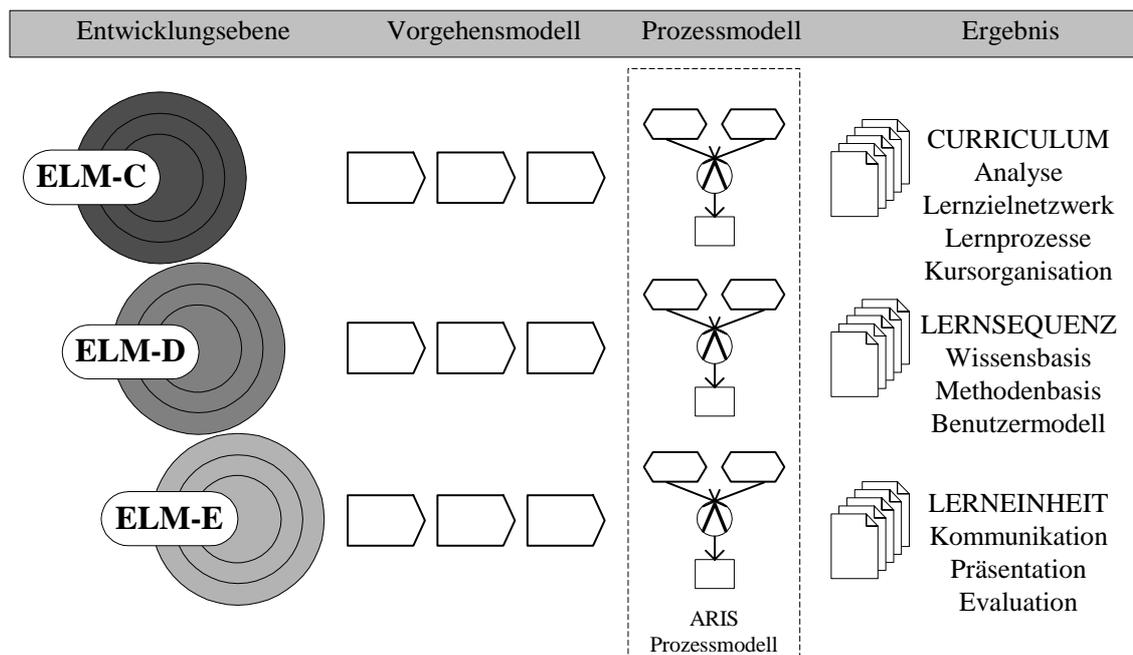


Abbildung 44: ELM-Entwicklungsebenen

Das Essener-Lern-Modell bietet als generisches Vorgehensmodell umfangreiche Unterstützungsfunktionen. In den Verhandlungs- und Anpassungsschritten jeder Ebene müssen Funktionen ausgewählt werden, die im weiteren Vorgehen genutzt werden. Diese Anpassung führt zu einem spezifischen Prozessmodell. Nach Durchführung der Schritte des Prozessmodells liegen Ergebnisse in Form von Spezifikationen, Anwendungen und Komponenten vor, die jeweils wieder evaluiert werden.

5.3 Architektur des Essener-Lern-Modells

Die in Abschnitt 2.4 beschriebenen Architekturen erfüllen die Anforderungen an ein umfassendes Vorgehensmodell nur partiell, da beispielsweise didaktische Ansätze häufig nur implizit modelliert werden. Somit ermöglichen sie lediglich eine technologieorientierte Beschreibung, die zur genauen Charakterisierung und Vergleichbarkeit nicht ausreicht. Daher wird für das Essener-Lern-Modell eine Architektur entwickelt, die diese Schwachpunkte behebt und eine genaue Vergleichbarkeit und Charakterisierung erlaubt. Weiterhin bietet die Architektur ein Rahmenkonzept zur Interoperabilität von Lerntechnologien (Abbildung 45).

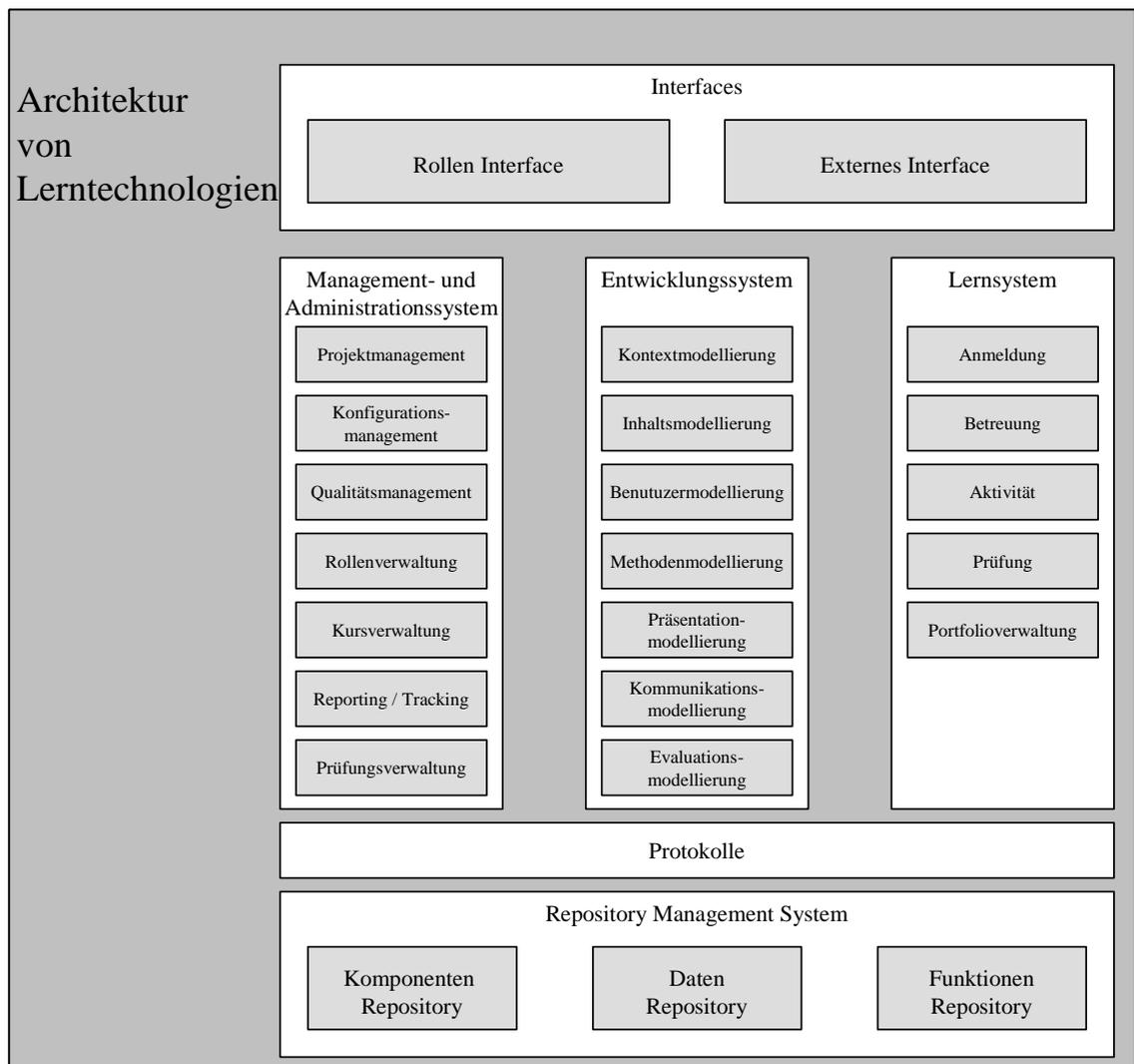


Abbildung 45: ELM-Architektur

Die vorgeschlagene Architektur unterscheidet vier Ebenen:

Die *Interface-Ebene* umfasst interne und externe Schnittstellen. Dabei wird durch das *Rollen-Interface* die Schnittstelle von Lerntechnologien zu Benutzern geschaffen und eine Anpassung an entsprechende Akteure vorgenommen, so dass die Sicht (Rechte, Präsentation) auf die verfügbaren Anwendungen rollenabhängig ist. Der Zugriff auf einzelne Systeme oder die Verwendung der Komponenten erfolgt über ein API. *Externe Interfaces* definieren Schnittstellen zu verschiedenen externen Applikationen (z. B. Personalentwicklungssysteme, betriebswirtschaftliche Anwendungssysteme), während interne Interfaces die Schnittstellen zwischen den einzelnen Anwendungen des Systems spezifizieren (z. B. Kursverwaltung zu Inhalten).

Die *Systemebene* enthält verschiedene Anwendungen, die in Management- und Administrationssysteme, Entwicklungssysteme und Lernsysteme unterteilt werden können. Ein System besteht dabei aus Subsystemen, die jeweils die Aktivitätsbereiche abdecken. Dabei

können die einzelnen Systeme unter Verwendung der bestehenden Komponenten adaptiert und kombiniert werden. Der Datenaustausch und Funktionsaufruf erfolgt über *Protokolle*.

Die jeweiligen Komponenten, Daten und Funktionen werden in einem *Repository Management System (RMS)* verwaltet und gespeichert. Unter einer Komponente versteht man dabei ein Objekt, das zur Generierung von Anwendungen verwendet wird (z. B. Benutzerverwaltung). Die Komponenten können durch einzelne Funktionen individuell angepasst und verwendet werden (z. B. Anmeldung, Profiländerung). Die Funktionen verwenden dabei entsprechende Datenmodelle (z. B. Benutzerprofil).

Abbildung 46 zeigt das Datenschema des Essener-Lern-Modells. ELM wird jeweils innerhalb von Projekten verwendet, die durch Projektziele abgegrenzt sind. Ein Projekt findet immer in einem organisationalen Kontext statt, für den ein spezifisches Prozessmodell angepasst wird. Der Kontext und das Prozessmodell bestimmen den Ablauf der Entwicklung von Lernobjekten.

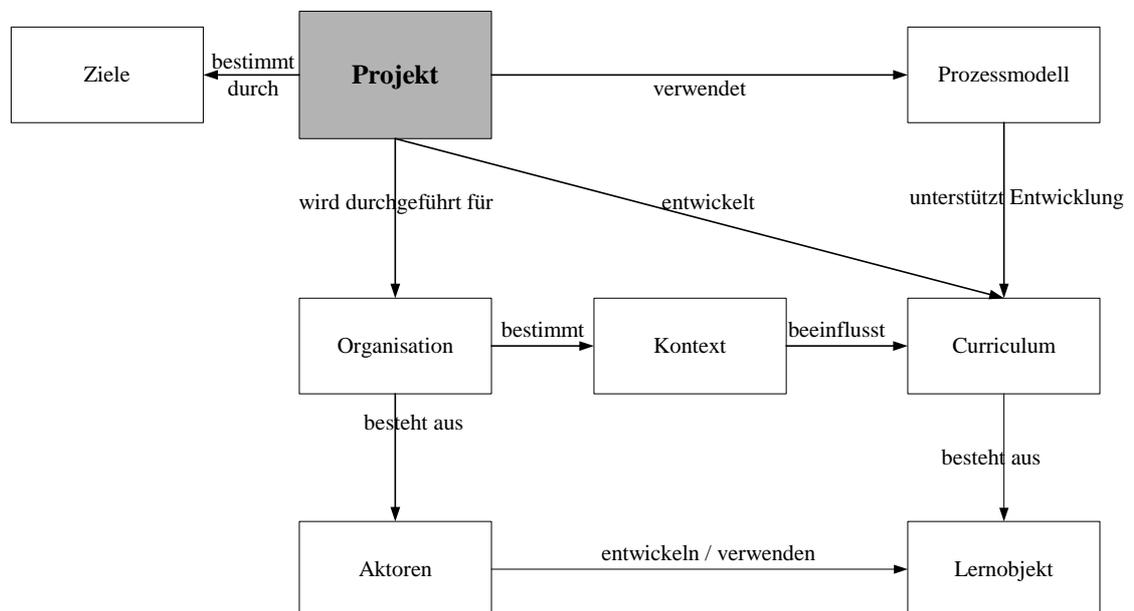


Abbildung 46: Datenschema der Top-Level-Entities

Die einzelnen Ausprägungen von Komponenten, Funktionen und Daten werden in den folgenden Abschnitten diskutiert; hier ist zunächst nur die abstrakte Architektur dargestellt.

Die vorliegende Architektur identifiziert die maßgeblichen Anwendungen und deren Interdependenzen. Es wird ein Rahmenkonzept zur Abbildung von Lernumgebungen geschaffen, das für eine Vergleichbarkeit einzelner Architekturen verwendet werden kann. Des Weiteren ist es Ausgangspunkt für die Interoperabilität von Anwendungen (siehe 4.1).

In den folgenden Abschnitten 5.4 bis 5.6 werden die Entwicklungsebenen des Essener-Lern-Modells beschrieben. Dabei werden jeweils die maßgeblichen *Funktionen* und Daten-schemata erläutert. Zur besseren Verständlichkeit werden nicht die vollständigen *Datenmodelle*, sondern jeweils eine schematische Darstellung und Erläuterung aufgeführt. Hinzu kommt die Beschreibung eines *Informationsmodells*, das die Kategorien und Attribute der Spezifikationen zusammenfasst. Ein Informationsmodell beschreibt die Inhalte einer Spezifikation, nicht aber die Implementierung [IEEE2001a]. Das bedeutet, dass individuelle Regeln, Ausprägungen oder Wertebereiche definiert werden können. So sind zum Beispiel keine einheitlichen Kompetenzdefinitionen zu finden, da diese meist organisationsspezifisch definiert werden. Somit muss für diese Kategorie eine Anpassung vorgenommen werden. Durch die Verwendung des Informationsmodells wird ein flexibles Modell vorgeschlagen, das wiederum projektspezifisch angepasst werden kann. Ferner wird zu jeder Spezifikation eine Anpassung des Dublin Core-Modells vorgenommen, das die Beschreibung von Inhalten, Verantwortlichkeiten und Versionen ermöglicht. Dadurch liegt jede Spezifikation in interoperabler Form vor und kann somit von anderen Systemen (z. B. Dokumentenmanagementsystemen, Computer Supported Cooperative Work-Systemen, Wissensmanagementsystemen) übernommen werden.

5.4 Vorgehensmodell zur Ausbildungsplanung (ELM-C)

ELM-C ist die Ebene zur Kontextanalyse, Ausbildungsplanung und Curriculumentwicklung. Bezogen auf die Architektur von Lerntechnologien (siehe 5.3) umfasst ELM-C Management- und Entwicklungsfunktionen:

Während der Projektinitiierung (siehe 5.4.1) werden die Rahmenbedingungen eines Projekts festgelegt. Bei der Kontextanalyse (siehe 5.4.2) wird das Umfeld der Lernumgebungen analysiert. Daraufhin folgt der Entwicklungsprozess im engeren Sinne (siehe 5.4.3). Auf dieser Ebene wird ein Curriculum in Form von Lernzielen erarbeitet und evaluiert. Abbildung 47 fasst die Prozesse und Aktivitäten aus ELM-C zusammen.

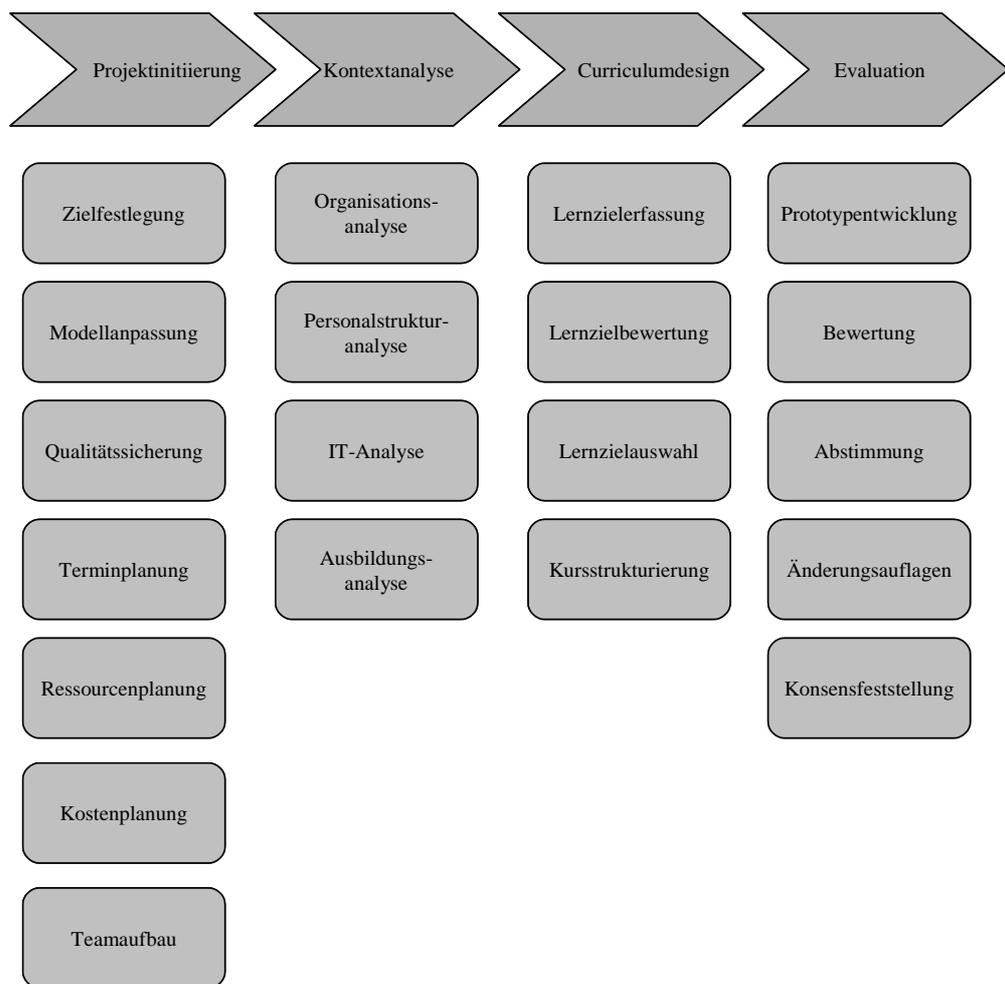


Abbildung 47: Wertschöpfungskette ELM-C

5.4.1 Projektinitiierung

Die Projektinitiierung beinhaltet die Aktivitäten des Projektmanagements, des Konfigurationsmanagements und der Qualitätssicherung.

Zunächst erfolgt eine *Zielfestlegung* zwischen Auftraggeber und Entwickler. In diesem Verhandlungsprozess werden zunächst die übergeordneten Zielsetzungen festgelegt, die die weitere Richtung des Projekts vorgeben. In diesem Zusammenhang werden primär taktische und operative Ziele der Entwicklung von Lernumgebungen betrachtet. Die Zielsetzung muss im Laufe der C-Ebene weiter detailliert werden, da Ergebnisse aus anderen Aktivitätsbereichen (z. B. aus der Kontextanalyse oder dem Curriculumdesign) Zielsetzungen modifizieren können. Diese Änderungen müssen dann jeweils in einer neuen Verhandlungsphase und Konsensfeststellung überarbeitet werden.

Eine erste *Modellanpassung* erfolgt im Anschluss an die Projektinitiierung. In einem Team (Projektverantwortliche, Lehrende, Lernende, Designer) wird das Vorgehensmodell auf die Bedürfnisse eines spezifischen Projekts angepasst. Dabei werden die einzelnen Aktivitäten von ELM analysiert; es wird bewertet, ob eine bestimmte Aktivität im Projekt durchzuführen

ren ist. Ist z. B. bereits ein detailliertes Curriculum implementiert, wird die Phase der Curriculumentwicklung verkürzt, so dass das Curriculum nur abgebildet werden muss. Des Weiteren kann die Terminologie der Aktivitäten und Spezifikationen an projekt- oder organisationsspezifische Sprachregelungen angepasst werden. Die Anpassung wird durch die Ergebnisse der Kontextanalyse noch weiter spezifiziert. Ebenso werden im Projektverlauf weitere Anpassungen vorgenommen, da jeweils in der Verhandlungsphase einer Aktivität ein Konsens über das weitere Vorgehen bestehen muss.

Die Bestimmung einer Methode zur *Qualitätssicherung* ist ebenfalls Teil der Projektinitiierung. Zielen werden jeweils Aktivitäten und Ergebnisse zugeordnet, so dass dadurch ein Rahmen für eine prozess- und produktorientierte Qualitätssicherung vorgegeben wird. Die Transparenz ist ein wichtiger Faktor für die Akteure des Entwicklungs- und Lernprozesses. Wie bereits in den vorhergehenden Kapiteln beschrieben, differiert die Qualität von Lernumgebungen wesentlich. Häufig ist es nicht ersichtlich, welche Merkmale ein Produkt aufweist. Daher werden durch ELM umfangreiche Spezifikationen zur Verfügung gestellt, die einerseits als Dokumentation zur Wartung oder Weiterentwicklung genutzt werden können, andererseits als Basis zur Benutzerinformation dienen. Durch die Einführung der Verhandlungsphasen, der Konsensfeststellung und der laufenden Bewertung ist ein interner Evaluationsrahmen vorgegeben. Weitere Qualitätsstandards (wie etwa ISO 9000 oder Kriterienkataloge) können in diesen Rahmen einbezogen werden. Somit wird eine laufende Evaluation durchgeführt, die in das Prozessmodell integriert ist.

Die Entwicklung computerunterstützter Lernumgebungen erfordert ein hohes Maß an Koordination zwischen Beteiligten unterschiedlicher Domänen. Mithilfe verschiedener Managementfunktionen wird die Aufbau- und Ablauforganisation unterstützt. Dieses kann einerseits durch die Einbindung externer Systeme, andererseits durch die eigenständige Unterstützung des Vorgehensmodells erfolgen. Im Essener-Lern-Modell werden Basisfunktionen des Projektmanagements unterstützt.

Teamaufbau: Die Beteiligung von Akteuren verschiedenster Fachrichtungen und deren Koordination erfordert eine eindeutige Planung und Steuerung. Es müssen zunächst die Kompetenzen der Beteiligten erfasst werden, um Aktivitäten und Aufgaben Akteuren mit entsprechender Qualifikation zuordnen zu können. Zudem müssen Verantwortlichkeiten der einzelnen Teilbereiche eindeutig festgelegt sein. Ergebnis dieser Planung ist die Zuordnung von Akteuren zu Teams. Diese Zuordnung kann auch zur Laufzeit des Projekts erfolgen, um die Flexibilität der Planung zu erhöhen.

Terminplanung/Ressourcenplanung: Ebenfalls begründet durch den hohen Koordinationsaufwand müssen die Zeiten und entsprechende Ressourcen den Aktivitäten zugeordnet werden. Hinzu kommt ein Vergleich der Soll- und Ist-Zeiten, die durch die Applikation erfasst und kontrolliert werden können. Dabei können Meilensteine spezifiziert werden, es kann jedoch auch eine zeitliche Zuordnung auf Aktivitätsebene erfolgen.

Kostenmanagement/Kostenplanung: Die Planung sowie die stetige Steuerung und Kontrolle der Entwicklungskosten von Lernumgebungen ist ein zentraler Bereich im Hinblick auf eine wirtschaftliche Durchführung der Entwicklung. In ELM kann ein Kalkulationsmodell und die dazugehörige Kalkulation spezifiziert werden. Alternativ ist auch die Referenzierung spezifischer Kostenplanungs- und Kostenabrechnungssysteme möglich.

Das zugehörige Informationsmodell ist in Tabelle 26 zusammengefasst:

Tabelle 26: Informationsmodell Projektdaten

| Kategorie | Beschreibung |
|---------------------------|--|
| Allgemeine Daten | |
| Dublin Core | Dublin Core-Elemente für Projektspezifikation: <ul style="list-style-type: none"> • Name des Projekts • Projektverantwortlicher • Inhalt des Projekts • Beschreibung des Inhalts • Verantwortlicher für Veröffentlichung der Projektspezifikation • Verantwortlicher für Inhalte der Spezifikation • Datum einer Änderung • Bereich des Projekts • Referenz zu Ergebnissen und weiteren Anwendungen/Spezifikationen • Quelle • Sprache • Referenz zu verwandter Ressource • Bereich • Rechte zur Nutzung und Verwendung der Projektspezifikation |
| Referenz | Referenz zu weiterem Informationsmodell |
| Zielsetzung | |
| Zielbezeichner | Bezeichner der Zielsetzung |
| Zielbeschreibung | Beschreibung eines Projektziels |
| Zielkategorie | strategisch, taktisch, operativ |
| Zielbereich | Aktivitätsbereich (z. B. Curriculum, Lernumgebung, Lerneinheit) |
| Ergebnisse | Beschreibung vereinbarter Ergebnisse (Spezifikationen, Produkte) |
| Projektaktivitäten | |
| Aktivität | Aktivitäten aus ELM, Reihenfolge, Restriktionen |
| Verantwortlicher | Person, verantwortlich für Projektaktivität |
| Mitarbeiter | Mitarbeiter einer Projektaktivität |
| Ergebnisse | Spezifikationen, Prototypen, Produkte |
| Externe | Auftragnehmer |
| Dauer | zeitliche Ressourcenzuordnung |
| Kapazität | Kapazität einer Ressourcenzuordnung |
| Status | Status/Statusänderung |
| Referenz | Referenz zu anderen Spezifikationen |

| Kosten | |
|---------------------------|--|
| Kalkulationsmodell | Beschreibung der Kostenkalkulation |
| Kalkulation | Kalkulation der Kosten |
| Verweise | Verweis auf Kostenrechneranwendung |
| Gesamtkosten | Volumen des Projekts |
| Bemerkungen | Anmerkungen zu Kostenfaktoren |
| Referenz | Referenz zu anderen Spezifikationen |
| Qualitätssicherung | |
| Methode | Methode der Qualitätssicherung (QS) |
| Aktivität | Projektaktivität als Subjekt der QS-Maßnahme |
| Ergebnis | Ergebnis als Subjekt der QS-Maßnahme |
| Aktor | Verantwortlicher der QS-Maßnahme |
| Art | Art der QS-Maßnahme: Dokumentation/Anforderung |
| Katalog | Referenz auf Kriterienkatalog |
| Bewertung | Bewertung der QS-Maßnahme |
| Änderung | Änderungsanforderungen der QS-Maßnahme |
| Referenz | Referenz zu weiteren QS-Spezifikationen |

5.4.2 Kontextanalyse

Die Kontextanalyse ist der Ausgangspunkt eines Projekts. Viele Lernumgebungsentwicklungen scheitern einerseits an der fehlenden Einbettung in einen organisationalen Kontext, indem komplexe Systeme entwickelt werden, ohne bereits vorher entwickelte Lösungen einzubeziehen. Andererseits bleiben bestehende Ressourcen ungenutzt, die in der Entwicklung von Lernumgebungen sinnvoll verwendet werden können. So führt z. B. die Einbettung von Inhalten aus Wissensmanagementsystemen zu einem wesentlich geringeren Aufwand bei der Erstellung von Lerninhalten. Des Weiteren können durch die Einbindung von Anwendungssoftware praxisnahe Lerninhalte ohne großen Aufwand generiert werden. Derartige Übungsfälle können aus dem betrieblichen Anwendungskontext extrahiert und verwendet werden, ohne neue Situationen, die häufig nicht an die Benutzerbedürfnisse angepasst sind, erstellen zu müssen. Ferner sollten bestehende Systeme zur Entwicklungsunterstützung (Hypertextsysteme, Autorensysteme, Programmierumgebungen) eingebunden werden. Ein weiterer Grund zur Durchführung einer Kontextanalyse ist die Überprüfung der Einsetzbarkeit verschiedener Konzepte von Lernumgebungen. Beispielsweise können in Organisationen mit Netzwerken geringer Bandbreite keine aufwendigen Streaming-Lösungen implementiert werden. Mithilfe der Kontextanalyse können demnach zahlreiche Problemstellungen schon in der Planungsphase erkannt und entsprechende Maßnahmen eingeleitet werden.

Das Datenschema in Abbildung 48 zeigt die maßgeblichen Objekte dieser Phase. Zunächst wird eine *Analyse der Organisations- und Personalstruktur* durchgeführt, bei der Organisationsziele, -philosophie, Produkte bzw. Dienstleistungen sowie verwendete Normen und interne Standards abgebildet werden. Diese bilden einen Rahmen für die Entwicklung der

Lerntechnologien. Sind z. B. Standards zur Präsentation (z. B. als Teil einer Corporate Identity-Richtlinie) oder zum internen Berichtswesen vorhanden, so müssen diese von Beginn an berücksichtigt werden. Hinzu kommt die Abbildung von Abteilungen, Teams und Personen. So werden einerseits die späteren Anwender identifiziert und charakterisiert, andererseits können Experten zur Unterstützung der Entwicklung herangezogen werden.

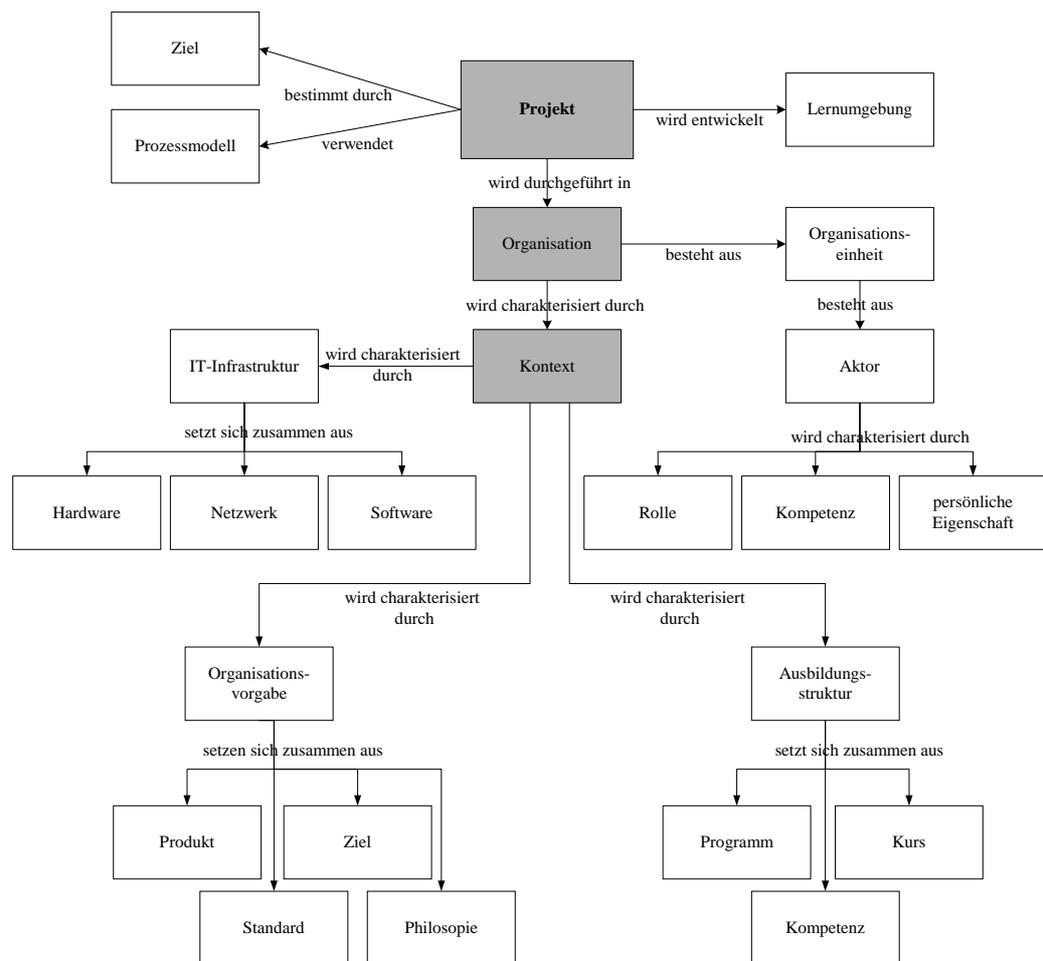


Abbildung 48: Datenschema Organisation/Kontext

Die *Analyse der IT-Infrastruktur* erfasst Hardware-, Software- und Netzwerkkomponenten. Somit werden notwendige Beschränkungen der technischen Umsetzung identifiziert. Weiterhin können Veränderungen bzw. notwendige Neuanschaffungen frühzeitig eingeleitet werden. Ferner wird analysiert, welche Komponenten (Software, Plug-Ins) nutzbar sind, um den Entwicklungsaufwand zu minimieren. Die Analyse ist jedoch nicht nur aus technischer Sicht nützlich, sondern schafft zudem einen Rahmen für die Entwicklung neuer Systeme. Bestehende Anwendungen, Betriebssysteme oder Benutzeroberflächen sollten berücksichtigt werden, um den Aufwand zum Erlernen der Systembedienung gering zu halten und eine einfache Benutzung sicherzustellen.

Abschließend wird die *Analyse der Ausbildungsstruktur* durchgeführt, die ebenfalls Rahmenbedingungen für die Entwicklung schafft. So werden bestehende Curricula in Katalo-

gen bzw. Taxonomien abgebildet, um neue Entwicklungen in diesen Kontext einbinden zu können. Des Weiteren werden spezifische Programme, Zertifikate und korrespondierende Kompetenzen abgebildet. Die Beschreibung von Kompetenzen kann bereits personenspezifisch durchgeführt werden, um das Vorwissen der späteren Lernenden zu identifizieren oder um initiale Testgruppen zu bilden. Am Ende dieser Phase wird eine neue Verhandlungsphase initiiert, in der festgelegt wird, welche strukturellen, technischen und curricularen Veränderungen angestoßen werden müssen.

Tabelle 27: Informationsmodell Kontextanalyse

| Kategorie | Beschreibung |
|---------------------------|---|
| Allgemeine Daten | |
| Dublin Core | Dublin Core-Elemente für Kontextanalyse |
| Referenz | Referenz auf weiteres Informationsmodell |
| Organisation | |
| Adressdaten | Name, Adresse, Telefon |
| Beschreibung | Beschreibung des Tätigkeitsbereichs |
| Philosophie | Philosophie/Strategische Ziele einer Organisation |
| Produkte | Services/Produkte einer Organisation, Verweis auf Katalog |
| Standards | verwendete Normen und Standards (intern/extern) |
| Abteilungen | Name, Bereich, Leiter, Personen |
| Teams | Name, Bereich, Leiter, Personen |
| Referenz | Referenz zu anderen Spezifikationen |
| IT-Analyse | |
| Hardware | Hardware-Ausstattung: Rechner, Peripherie |
| Software | Beschreibung von Applikationen: Name, Version, Betriebssystem, Nutzergruppen, Bemerkungen (z. B. zu Lizenzen) |
| Plug-Ins | unterstützte Formate (Name, Version, Extension, Applikation) |
| Netzwerk | System, Bandbreite, Protokolle, Einschränkungen, Nutzergruppen |
| Autorensysteme | Name, Beschreibung, Version |
| Entwicklungsumgebungen | Name, Beschreibung, Version |
| Referenz | Referenz zu anderen Spezifikationen |
| Ausbildungsanalyse | |
| Curriculum | Beschreibung eines Curriculums |
| Programme | Ausbildungsprogramme, Verweis auf Lernziele |
| Kompetenzen | Name, Beschreibung, Inhalt einer Kompetenz |
| Zertifikate | Zertifikate für Programme/Kompetenzen |
| Lernumgebungen | Beschreibung, Inhalt, Methode, Lernziele |
| Lernobjekte | Bestehende Objekte (siehe Lernobjektbeschreibung) |
| Personen | Name, Rolle, Kompetenzen |
| Referenz | Referenz zu anderen Spezifikationen |

Die Kontextanalyse schafft die Rahmenbedingungen für den weiteren Entwicklungsprozess. Somit werden die Aktivitäten und zu entwickelnden Spezifikationen und Ergebnisse eingegrenzt, so dass redundante Entwicklungen vermieden werden und die Entwicklung kontextspezifisch verläuft. Die Spezifikationen ist im Informationsmodell (Tabelle 27) zusammengefasst.

5.4.3 Curriculumdesign/Evaluation

Die Aktivitätsbereiche des Curriculumdesign und der Evaluation sind der Ausgangspunkt für die Kursentwicklung. In diesen Aktivitäten werden Ausbildungsprogramme unter Verwendung von Lernzielen spezifiziert und evaluiert.

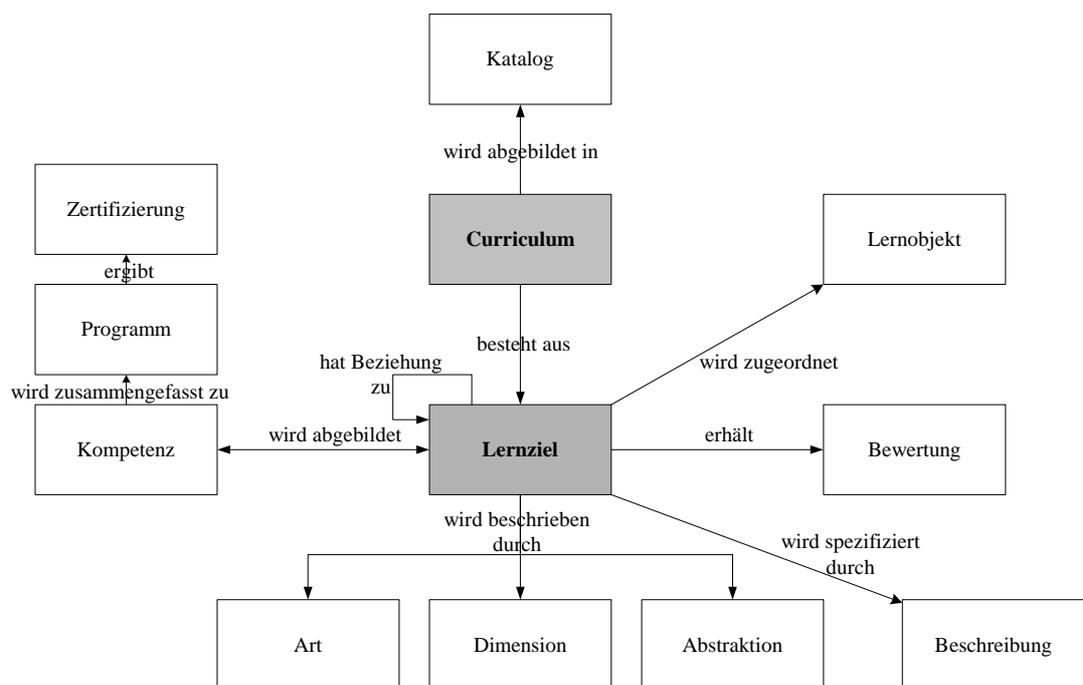


Abbildung 49: Datenschema Curriculum

Der Aktivitätsbereich der *Curriculumentwicklung* hat die Aufgabe, ein detailliertes Ausbildungskonzept durch die Spezifikation von Programmen, Curricula, Kursen und Kompetenzen zu entwickeln. Es ist offensichtlich, dass ebenso nur einzelne Kurse spezifiziert werden können, wenn bereits ein umfassendes Curriculum besteht. Wie bereits in Abschnitt 2.3.3 erläutert, kann ein Curriculum durch ein System von Lernzielen spezifiziert werden. Ausgangspunkt ist die Erfassung von Lernzielen, die zunächst in Form eines Brainstorming durchgeführt wird. Die Ergebnisse werden daraufhin nach Art, Dimension und Abstrakti-

onsgrad gegliedert. Aufgrund der Abstraktionsniveaus (Richtziel, Grobziel, Feinziel, vgl. 2.3.3.1) ergibt sich bereits eine hierarchische Struktur. Hinzu kommen frei spezifizierbare Relationen, die zu einem Netzwerk von Lernzielen führen. Die Beschreibung und Kategorisierung von Lernzielen sollte in einem Team aus Domänenexperten, Entwicklern, Lehrenden und Lernenden erfolgen.

Im Anschluss an die Sammlung, Formulierung und Strukturierung der Lernziele erfolgt eine *Bewertung*, um Ziele aufgrund ihrer Relevanz zu priorisieren. Die Priorisierung kann anhand folgender Kriterien in Anlehnung an [Möll1973] erfolgen:

1. Kriterium der individuellen Anforderungen
2. Kriterium der organisationalen Anforderungen
3. Kriterium der Konsistenz
4. Kriterium der Kontrollierbarkeit
5. Kriterium der aktuellen und zukünftigen Relevanz
6. Kriterium der Erreichbarkeit durch Personalentwicklungsmaßnahmen

Diese Kriterien werden projektspezifisch angepasst und operationalisiert.

In einem Verhandlungsprozess erfolgt schließlich die Auswahl der Lernziele, eine grobe Strukturierung der korrespondierenden Lernobjekte sowie die Zuordnung von Lernzielen zu Kursen bzw. Lerneinheiten. Diese Struktur wird in ELM-D weiterentwickelt, die initiale Zuordnung dient insbesondere der Übersichtlichkeit.

Die *Evaluation* der ersten Ebene erfolgt im Anschluss an die Curriculumentwicklung. Es wird ein prototypischer Katalog entwickelt, der das Curriculum bzw. die Lernziele abbildet und mit Lernobjekten verknüpft. Die Ergebnisse von ELM-C (Prototyp und Spezifikationen, Tabelle 28) werden bewertet und Änderungsaufgaben spezifiziert, bis ein Konsens erreicht wird, der als Meilenstein der ersten Ebene gesehen werden kann.

Tabelle 28: Informationsmodell Curriculum

| Kategorie | Beschreibung |
|--------------------------------|---|
| <i>Allgemeine Daten</i> | |
| Dublin Core | Dublin Core-Elemente für Curriculum |
| Referenz | Referenz auf weiteres Informationsmodell |
| <i>Lernziele</i> | |
| Zielbezeichnung | Bezeichnung eines Lernziels |
| Dimension | Dimension eines Ziels: kognitiv, affektiv, psychomotorisch, sozial |
| Komplexität | Komplexität eines Lernziels |
| Art | Art eines Lernziels: Fakten/kontextfreie Regeln, kontextabhängige Regeln, Problemlösung, Gestalt- und Mustererkennung, komplexe Situationen |
| Abstraktion | Abstraktionsniveau eines Lernziels: Grobziele, Richtziel, Feinziel |
| Beziehung | Beziehung zu anderen Lernzielen |
| Katalog | Einordnung in einen Katalog von Lernzielen |
| Status | Status eines Lernziels |
| Bewertung | Bewertung eines Lernziels |
| Lernobjekt | Referenz auf Lernobjekt |
| Referenz | Referenz auf andere Spezifikationen |
| <i>Programme</i> | |
| Beschreibung | Beschreibung eines Programms |
| Lernziele | Lernziele eines Programms |
| Kompetenz | Kompetenzen eines Programms |
| Zertifizierung | Zertifizierung für ein Programm |
| Referenz | Referenz zu anderen Spezifikationen |
| <i>Kompetenzen</i> | |
| Kompetenz | Name, Beschreibung, Inhalt einer Kompetenz |
| Lernziel | Referenz zu Lernziel |
| Zertifikate | Zertifikate für Programme/Kompetenzen |
| Lernobjekte | Referenz zu Lernobjekt |
| Referenz | Referenz zu anderen Spezifikationen |

5.4.4 Zusammenfassung ELM-C

In ELM-C werden Aktivitäten des Projekt-, Konfigurations- und Qualitätsmanagements bearbeitet. Dabei werden verschiedene Spezifikationen entwickelt, die einerseits als Projektberichte dienen, andererseits von ELM in weiteren Phasen verwendet werden. Zu jeder Spezifikation in ELM liegen zudem Beschreibungsdaten nach Dublin Core vor, die in anderen Applikationen verwendet werden können.

Zentrale Ergebnisse sind die Kontextanalyse und ein Prototyp des Curriculums, das als Lernzielnetzwerk abgebildet wird und Verknüpfungen zu Lernobjekten enthält. Durch die laufende Evaluation und Konsensfeststellung wird die Qualität der Projektziele und der allgemeinen Ziele sichergestellt.

Die Spezifikation der Informationsmodelle bezieht einerseits bestehende Standards ein (z. B. Kompetenzen), umfasst aber auch Metadaten für Teilbereiche, die derzeit nicht von Standardisierungsinitiativen bearbeitet werden. ELM bietet ein durchgängiges Informations- und Datenmodell, das als Vorlage für eine weitere Standardisierung dienen kann.

5.5 Vorgehensmodell zur Entwicklung von Lernsequenzen (ELM-D)

ELM-D umfasst die Entwicklung von Lernsequenzen in vier Phasen (Abschnitte 5.5.1 bis 5.5.4).

Während der *Wissensakquisition* werden die Lerninhalte der Lernumgebung im Detail spezifiziert. Parallel werden die späteren Benutzer soweit möglich charakterisiert, um bereits eine initiale Anpassung vornehmen zu können. Der entscheidende Aspekt, der diese Phase von anderen Vorgehensmodellen abhebt, ist die Modellierung und Anwendung didaktischen Methodenwissens zur Generierung von Lernumgebungen. Dabei werden Methoden vorgeschlagen und Templates zur Verfügung gestellt, die die Implementierung spezifischer Methoden vereinfachen.

Diese Aktivitäten führen zu einem Designmodell der Lernumgebung (Abbildung 50), das Inhalte, Struktur, Methoden und Aktoren enthält. Ein Prototyp dieses Modells wird zum Abschluss dieser Ebene generiert und evaluiert.

Abbildung 51 zeigt abschließend die Wertschöpfungskette und die maßgeblichen Aktivitäten dieser Ebene.

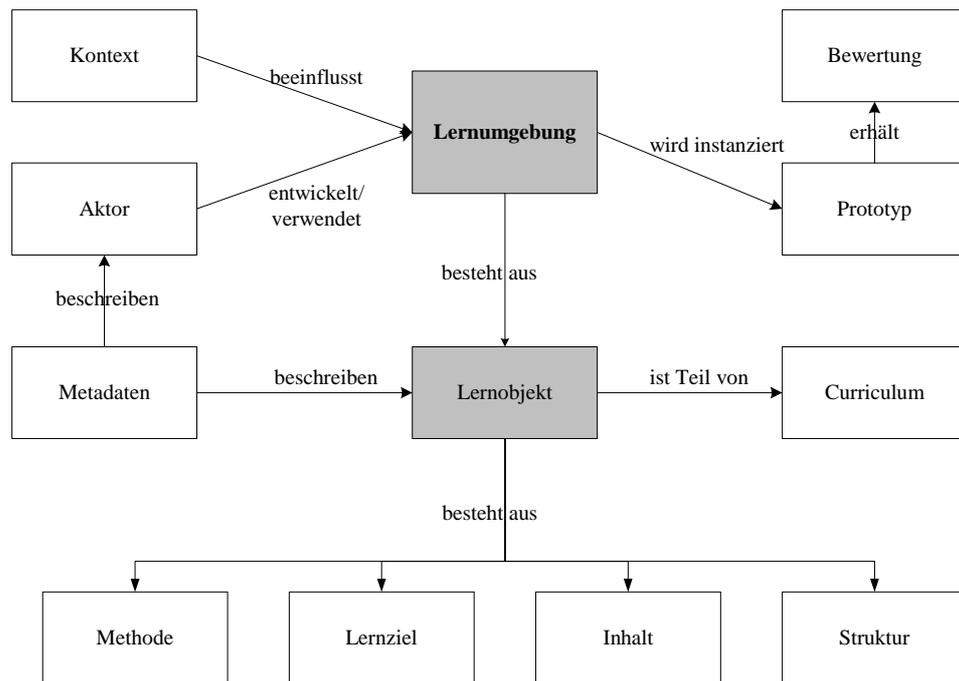


Abbildung 50: Datenschema Lernumgebung

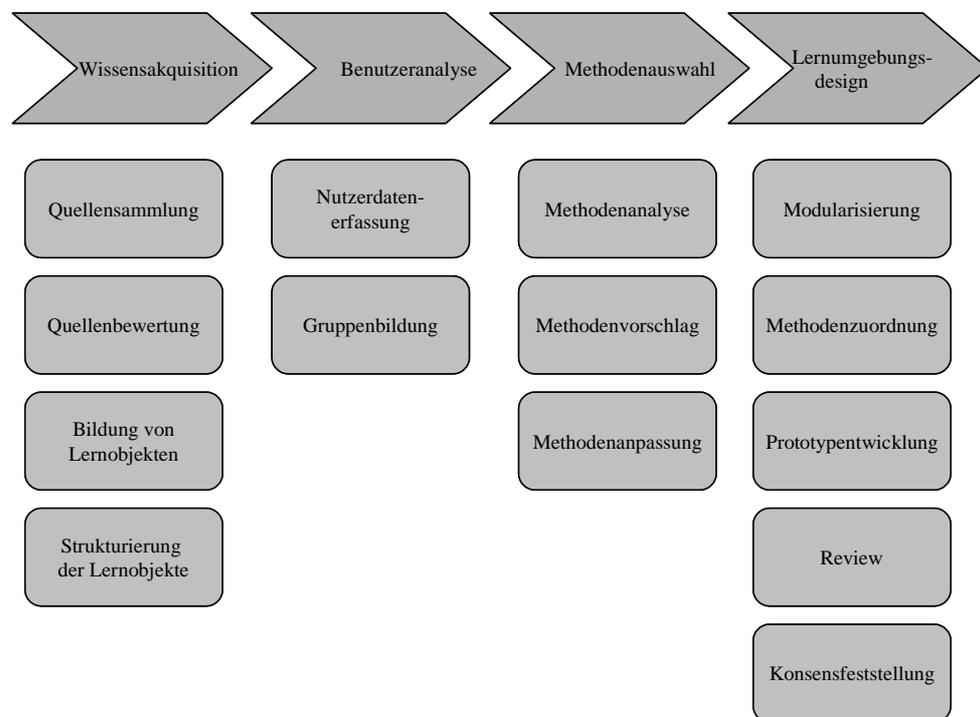


Abbildung 51: Wertschöpfungskette ELM-D

5.5.1 Wissensakquisition

Innerhalb des Aktivitätsbereichs der *Wissensakquisition* werden in einem Team aus Domänenexperten, Entwicklern, Lehrenden und Lernenden die Lerninhalte einer Lernumgebung recherchiert und im Detail festgelegt. Dabei werden zunächst Quellen zur Informationsrecherche erfasst, bewertet und einem Inhaltsobjekt zugeordnet. Die Bewertung erfolgt über die Faktoren Korrektheit, Einbettung in den Kontext, Aktualität, Relevanz und Kosten.

Aufgrund dieser Informationssammlung werden Lernobjekte gebildet, die sich in die Aggregationsebenen *Kurs*, *Zusammengesetzte Lerneinheit* und *Lerneinheit* gliedern. Die Lernobjekte werden generiert und strukturiert, indem Beziehungen zwischen Lernobjekten abgebildet und Lernziele zugeordnet werden. Durch diese Funktionen entsteht eine Kursstruktur. Bei der Generierung der Lernobjekte wird für jedes Lernobjekt ein Metadatensatz angelegt. Die Lernobjekte der Ebenen *Zusammengesetzte Lerneinheit* und *Lerneinheit* können dabei einen Großteil der Metadaten der jeweils übergeordneten Ebene übernehmen. Zudem werden verschiedene Metadaten automatisch generiert:

- *Allgemein*: Der Identifier wird automatisch generiert, ebenso werden Strukturen und Aggregationslevel aus der ELM-Spezifikation übernommen. Des Weiteren werden Katalogeinträge für die hierarchischen Beziehungen vorgenommen. Ein initialer Katalog wird unter Verwendung des Curriculums erstellt, der dann jeweils erweitert werden muss.
- *Lebenszyklus*: Soweit ein Lernobjekt von einem Akteur oder von Akteuren des Teams erstellt wurde, werden bei Änderung der Metadaten automatisch Einträge zu Version, Status und Verantwortlichen eingetragen. Die Versionsverwaltung kann nach Festlegung eines Versionsschemas automatisch generiert werden.
- *Meta-Metadaten*: Es wird jeweils die aktuelle, von ELM unterstützte Version von LOM eingetragen. Aktualisierungen werden jeweils automatisch erfasst, d. h., sobald z. B. eine neue LOM-Version implementiert wird, werden bei einer Anpassung auch die Werte der Meta-Metadaten aktualisiert.
- *Technische Metadaten*: Systemanforderungen werden jeweils vom Autoren spezifiziert und für hierarchische Lernobjekte vererbt.
- *Pädagogische Metadaten*: Die Einträge werden aus der ELM-Methodenspezifikation übernommen. Der Eintrag *Semantic Density*, der das Verhältnis der Komplexität der Inhalte zum Umfang erfassen soll, wird nicht verwendet, da es sich nicht um ein operationalisierbares und nur subjektiv vergleichbares Attribut handelt.
- *Rechte*: Rechte müssen jeweils spezifiziert werden. Als Default werden rechtliche Bestimmungen der Projektspezifikation übernommen.
- *Beziehung*: Beziehungen beschreiben Reihenfolge- und sachlogische Zusammenhänge. Hierarchische Beziehungen werden automatisch generiert, weiterhin werden Verknüpfungen zu Quellen automatisch eingetragen.
- *Erläuterung*: Erläuterungen werden jeweils von übergeordneten LOM übernommen und können erweitert werden.
- *Klassifikation*: Es steht ein eigenständiges Werkzeug zur Erstellung von Taxonomien zur Verfügung (ELM-Catalog, siehe 6.1.2), das die Eingabe von Klassifikationen erleichtert. Daneben können Standard-Klassifikationen importiert werden.

Es sei darauf hingewiesen, dass die Implementierung von Klassifikationen und Katalogen integraler Bestandteil der Verwendung von Metadaten ist. Daher wird bereits das Curriculum in Form eines Kataloges spezifiziert sowie um zugehörige Lernobjekte erweitert. Dennoch sollten inhaltliche Klassifikationen zusätzlich erstellt werden, um eine genauere Einordnung durchführen zu können.

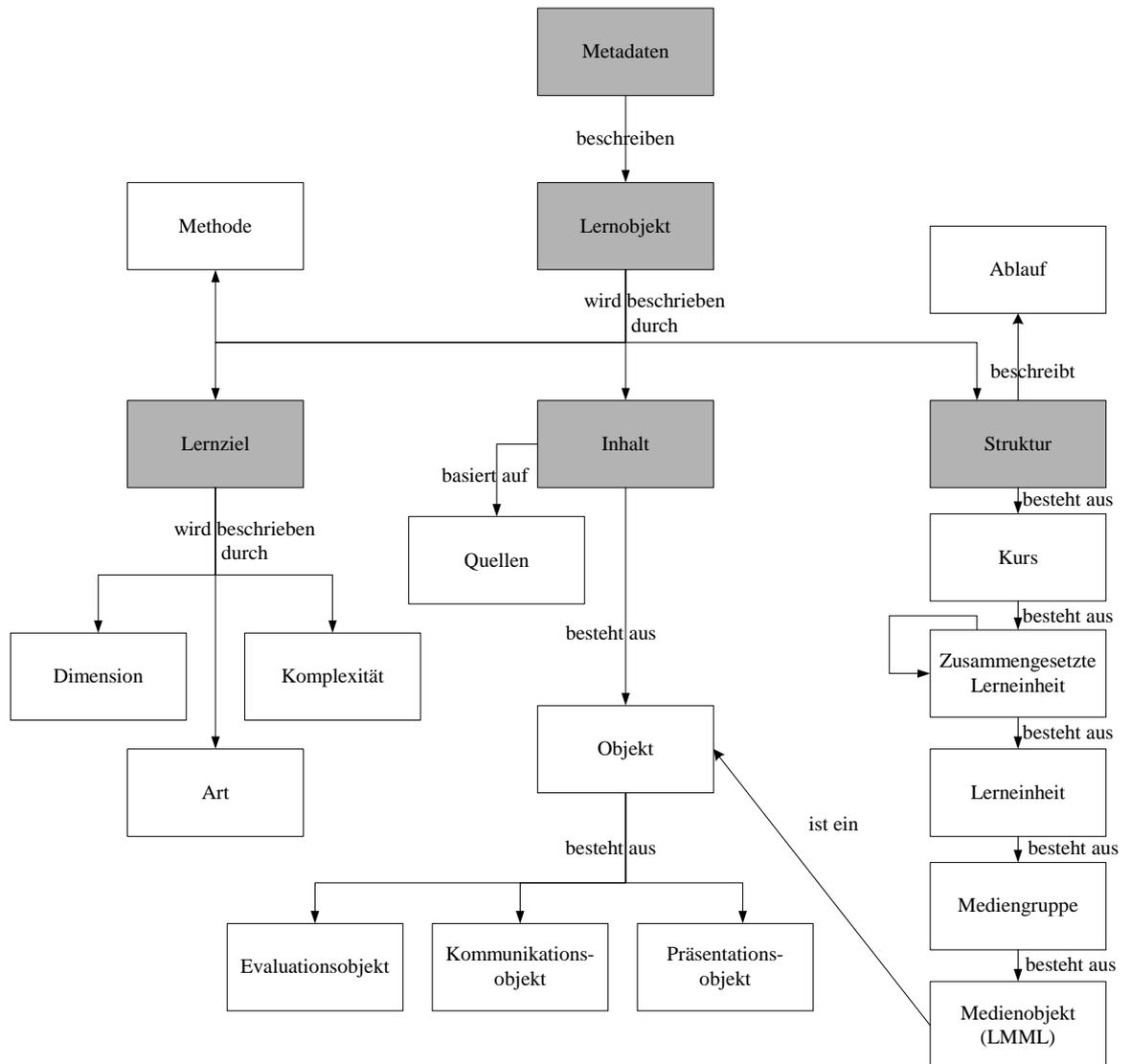


Abbildung 52: Datenschema Lernobjekt

Die Datenstruktur der Lernobjekte in dieser Aktivität ist in Abbildung 52 dargestellt, während das Informationsmodell (Tabelle 29) die Spezifikation eines Lernobjektes zusammenfasst. Lernobjekte der Aggregationsebene *Kurs* werden zudem in der Strukturspezifikation des Content Structure Format (CSF) beschrieben. Weiterhin werden korrespondierende Lernziele und Methoden (siehe 5.5.3) referenziert.

Bei der Instanziierung eines Lernobjektes werden Instanzdaten hinzugefügt. Sie umfassen zusätzliche Daten (Zeit und Ort eines Kurses bzw. einer Lerneinheit sowie die teilnehmenden Akteure), die bei einer Wiederverwendung nicht genutzt werden können (vgl. Kapitel 6).

Tabelle 29: Informationsmodell Lernobjekt

| Kategorie | Beschreibung |
|-------------------------|---|
| Allgemeine Daten | |
| Dublin Core | Dublin Core-Elemente zur Lernobjektspezifikation |
| Referenz | Referenz auf weiteres Informationsmodell |
| Beschreibung | |
| LOM | Beschreibung nach LOM, Version |
| Struktur | |
| CSF | Beschreibung nach SCORM-CSF |
| Aggregation | Aggregationsniveau: Kurs, Zusammengesetzte Lerneinheit, Lerneinheit |
| Referenz | Referenz auf weiteres Strukturmodell |
| Referenzen | |
| Kontext | Referenz auf Kontextmodell |
| Lernziele | Lernziele |
| Methoden | Referenz auf Methodenbeschreibung |
| Instanzdaten | |
| Zeit | zeitliche Beschreibung einer Instanz: Dauer, Intervall |
| Raum | Räumliche Beschreibung einer Instanz: Verteilung |
| Akteure | Referenz auf teilnehmende Akteure, Rolle |

Ergebnis dieses Aktivitätsbereichs ist die Spezifikation einer Kursstruktur, der Lernobjekte und der initialen Metadaten. Diese werden innerhalb der Lernobjektspezifikation zusammengefasst.

5.5.2 Benutzeranalyse

In der Benutzeranalyse werden die vorgesehenen oder bereits angemeldeten Benutzer charakterisiert. Wie bereits bei der Beschreibung der PAPI und LIP (siehe 4.4.4) erwähnt wurde, muss zu dieser Spezifikation ein Sicherheitskonzept zur Zugriffskontrolle auf persönliche Daten bestehen. ELM verfolgt das Konzept der benutzergesteuerten Autorisierung: Benutzer autorisieren den Lehrenden, dass ihre Daten erfasst und, eventuell anonymisiert, verwendet werden dürfen. Ist dies nicht der Fall, so werden die Benutzerdaten bis auf die persönlichen Anmeldedaten nicht erfasst.

Bei einer positiven Autorisierung werden Daten über Vorwissen, persönliche Ziele und Präferenzen erfasst, die an Lerneinheiten übergeben werden können. Es ist Aufgabe des Entwicklers, Lerneinheiten individuell anzupassen. Eine Übersicht über die erfassten Daten wird in Tabelle 30 gegeben.

Tabelle 30: Informationsmodell Aktoren

| Kategorie | Beschreibung |
|--------------------------|---|
| Allgemeine Daten | |
| Dublin Core | Dublin Core-Elemente für Aktorenspezifikation |
| Referenz | Referenz auf weiteres Informationsmodell |
| Persönliche Daten | |
| Name, Adresse | Name, Adresse (vcard) |
| Rolle | Beschreibung einer Rolle: Lernender, Lehrender, Entwickler, Manager, Domänenexperte, Mediendesigner |
| Team | Zuordnung zu einem Team |
| Teambeschreibung | Kriterien der Teameinteilung, Stereotypen, Rollen |
| Vorwissen | |
| Abschlüsse | Erworbene Abschlüsse |
| Programme | absolvierte Programme |
| Kompetenzen | erworbene Kompetenzen, erwartete Kompetenzen |
| Lernziele | absolvierte Lernziele |
| Methoden | Erfahrungen zu Lernmethoden |
| Lernschritte | Verweis auf absolvierte Lernschritte (Tracking) |
| Referenz | Verweis auf weitere Qualifikationsnachweise |
| Ziele | |
| Abschlüsse | angestrebte Abschlüsse |
| Programme | angestrebte Programme |
| Kompetenzen | angestrebte Kompetenzen |
| Lernziele | angestrebte Lernziele |
| Methoden | Erfahrungen zu Lernmethoden |
| Referenz | Verweis auf weitere Zielsetzungen |
| Präferenzen | |
| Kontext | Beschreibung des Kontextes |
| Methoden | bevorzugte Methoden |
| Präsentation | bevorzugte Präsentationsanwendungen |
| Kommunikation | bevorzugte Kommunikationsanwendungen |
| Lernstil | Cognitive Style, Learning Style |
| Lerntyp | Converger, Assimilator, Accomodator, Diverger |
| Referenz | Verweis auf weitere Zielsetzungen |
| Motivation | intrinsisch/extrinsisch, Beschreibung |
| Sicherheit | |
| Zugriffsrechte | Zugriff auf Aktorendaten (nach Rolle, Person, Team) |
| Verschlüsselung | Schlüssel, zu verschlüsselnde Informationen |
| Referenz | Referenz auf Sicherheitskonzept |

Die Daten der Benutzer können, soweit es das Sicherheitskonzept und die Autorisierung erlaubt, ebenfalls in andere Anwendungen übernommen werden. So können z. B. Personalentwicklungssysteme Informationen über Lernprozesse, Programme und Abschlüsse direkt übernehmen. Vorrangiges Ziel dieser Analyse ist jedoch die Charakterisierung der Benutzer zum Ziel der Benutzeradaptation. Die Anpassung an den Benutzer kann auf verschiedene Arten erfolgen: Entwickler können in Lernumgebungen auf bestimmte Benutzercharakteristiken zurückgreifen. So kann zum Beispiel ein Lernweg an die Vorkenntnisse und Lernpräferenzen des Benutzers adaptiert werden. Weiterhin kann dieses Modell auch nur zur Information der Lehrenden dienen, die diese Information zur spontanen Anpassung ihrer Beratung nutzen. Die Adaptationsmechanismen werden dabei nicht von ELM vorgegeben, vielmehr dient das Informationsmodell als Anhaltspunkt für Lehrende wie Entwickler, welche Benutzerdaten zur Anpassung verwendet werden können.

5.5.3 Methodenmodellierung

Ziel der *Methodenmodellierung* ist die Abbildung didaktischen Wissens zur Wiederverwendung und Anpassung von Methoden an Lerninhalte und Aktoren. Dabei werden keine präskriptiven Regeln im Sinne des Instruktionsdesign vorgegeben; vielmehr sind die Modelle von Methoden eine Sammlung von Erfahrungen und Vorlagen, die zur Erweiterung des Methodenspektrums von Lehrenden und Entwicklern verwendet wird.

Das Qualitätskriterium der Wiederverwendbarkeit bezieht sich in den meisten Fällen [vgl. LTSC2001c] auf die Wiederverwendung von Lerninhalten. Dazu wird ein Lerninhalt in Lernobjekte aufgeteilt, damit auch Teile eines Kurses in einem anderen Kontext genutzt werden können. Die Wiederverwendbarkeit didaktischer Methoden und damit auch didaktischen Wissens wird nur selten diskutiert [vgl. Kope2001]. Die formale Modellierung von Lernmethoden verfolgt daher folgende maßgebliche Zielsetzungen:

- *Beschreibung einer Methode*: Eine Lernmethode wird so abgebildet, dass weitere Benutzer die notwendigen Arbeitsschritte, eingesetzte Informationsobjekte und Erfahrungen nachvollziehen können. Die formale Beschreibung ermöglicht es, Methoden vergleichbar zu machen. Eine rein textuelle, nicht formale Beschreibung führt insbesondere durch eine unterschiedliche Terminologie zu missverständlichen Interpretationen (z. B. bezüglich des Kontextes der Methode) [vgl. AdBP2000a].
- *Vereinfachte Konzeption und Implementierung*: Das Modell einer Methode beschreibt umfangreiche Aspekte einer Lernmethode. Diese Beschreibungen können als Implementierungshilfe angesehen werden. So ist das Methodenmodell in einzelne Phasen gegliedert, die als Basis für Navigation oder eine sachlogische Abfolge eines Kurses dienen können. Weiterhin werden Präsentations-, Kommunikations- und Evaluationsobjekte vorgeschlagen, die automatisch generiert werden können. Wird zum Beispiel für eine Lernmethode die Verwendung eines Forums oder Chats vorgeschlagen, so kann der Autor diese Kommunikationsobjekte automatisch für eine Lernumgebung generieren.
- *Verwendung von Templates*: Für eine Lernmethode können Templates angelegt werden, die wiederverwendbar sind. Ein Template kann die sachlogische Abfolge einer

Methode beinhalten oder auch Designaspekte einbeziehen. Damit dienen Templates als Hilfsmittel für Autoren.

- *Wissensbasis*: Ein Methodenmodell umfasst Hinweise über Erfahrungen, die mit einer Methode in einem spezifischen Kontext erworben wurden. Damit dient das Modell als Hilfsmittel für Autoren zur Selektion einer Lernmethode. Im Essener-Lern-Modell können sowohl Hinweise und Anmerkungen als auch Evaluationen zu einer Methode erfasst werden. Dadurch haben Autoren die Möglichkeit Probleme und Schwierigkeiten beim Einsatz einer Methode a priori zu vermeiden.

Die Beschreibung einer Methode besteht aus den Hauptelementen Setting, Phasen, Präsentation, Kommunikation und Evaluation (Abbildung 53).

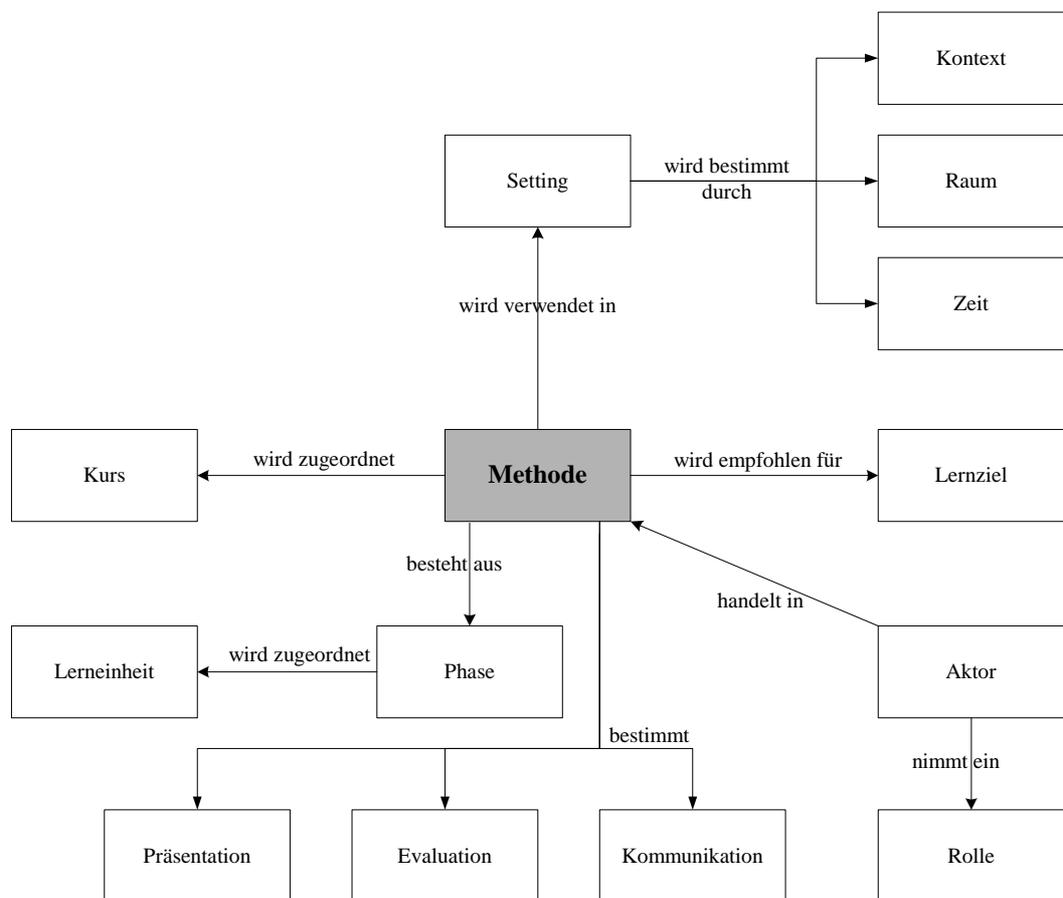


Abbildung 53: Datenschema Methode

Das *Setting* charakterisiert den Kontext, in dem eine Methode angewendet werden kann oder wurde. Dabei kann auf Erfahrungen in einem bestimmten Kontext verwiesen werden. Der Kontext beschreibt die Umgebung, in der eine Methode angewendet wurde. Dabei wird in ELM auf die Kontextspezifikation verwiesen, die organisationale, personale, technologische und ausbildungsspezifische Aspekte abbildet. Als Übersichtsattribute werden die räumliche (Verteilung) und zeitliche (Dauer) Gestaltung des Einsatzgebiets beschrieben. Hinzu kommt die Spezifikation, für welche Art von Lernzielen bzw. in welchen Kur-

sen eine Methode eingesetzt wurde. Durch diese Beschreibung kann eine Methode ex-post genauer beurteilt werden. Sind die Einsatzgebiete hinreichend ähnlich, so bietet sich der Einsatz dieser Methode in neuen Entwicklungen an. Dazu müssen jedoch weitere Vergleichskriterien projektspezifisch definiert werden. Durch diese Beschreibung kann der Erfolg oder Misserfolg einer Methode in verschiedenen Kontexten als Historie nachvollzogen werden und dient somit den Lehrenden als Entscheidungshilfe.

Die Modellierung einer Methode erfolgt durch die *Spezifikation von Phasen*. Eine Phase wird dabei jeweils einer Lerneinheit zugeordnet und beschreibt die Aktivität innerhalb einer Methode, wie zum Beispiel die Erarbeitung von Inhalten, Diskussionen, Vorträgen oder praktischen Übungen (vgl. 2.3.8). Neben einer textuellen Beschreibung enthält eine Phase Hinweise für Aktoren. So wird die Rolle des Lehrenden, des Lernenden und des Systems näher erläutert. Es werden Hinweise erfasst, wie ein Lehrender z. B. die Rolle eines Moderators bei der Methode des Scaffolding [Vygo1978] ausfüllt. Es wird spezifiziert, ob aktives oder passives Verhalten vorherrscht und ob von einem Lernenden aktives, passives, problemlösendes oder bearbeitendes Verhalten erwartet wird. Weiterhin wird die Anordnung und Reihenfolge der Phasen beschrieben. Einerseits kann eine lineare Reihenfolge, andererseits können weitere Sequenzoperatoren definiert werden, die eine flexiblere Gestaltung zulassen. So können Phasen frei angeordnet sein, d. h. der Lernende kann frei durch die Lernobjekte navigieren. Ferner können Blöcke von Phasen gruppiert werden, die zwar nacheinander bearbeitet werden sollten, aber innerhalb derer die Navigation freigestellt ist.

Methoden implizieren häufig bestimmte *Präsentationsformate*. Daher werden in jeder Phase Hinweise zu Präsentationsmöglichkeiten gegeben. Das bedeutet, dass Präsentationstypen und -anwendungen für eine Methode vorgeschlagen werden. So kann z. B. Video-Streaming als Teil der Anchored Instruction-Methode genutzt werden. Weiterhin werden Hinweise zur Verwendung dieser Präsentationsanwendungen gegeben, wie z. B. Installationshinweise, die den Einsatz vereinfachen.

Ein wichtiger Bestandteil des Methodenmodells ist die Beschreibung der *Kommunikation* innerhalb einer Methode. Hier werden Interaktionen detailliert beschrieben (Anwendung, Verlauf, Richtung, Rolle der Aktoren). Die genaue Beschreibung ist von großer Bedeutung, da gerade bei Interaktionen bestimmte Betreuungsfaktoren maßgeblich sind. So ist die Diskussionsleitung immer mit einem neutralen Standpunkt verbunden; der Diskussionsleiter nimmt die Rolle eines Moderators ein. Dabei sollte zu jedem Interaktionstyp (synchron, asynchron, unidirektional, bidirektional) eine entsprechende Anwendung verfügbar sein (Chat, Forum, News, E-Mail etc.), die in eine Lernumgebung eingebunden werden kann.

Zum Ende jeder Phase kann eine *Evaluation bzw. Leistungsüberprüfung* erfolgen. Diese kann in Form einer Prüfung oder Übung durchgeführt werden. Die verschiedenen Formate werden ebenfalls spezifiziert und mit entsprechenden Anwendungen (Fragebogengenerator, Anwendung als Übungsumgebung) verknüpft.

Die Beschreibung einer Methode wird in einer Verhandlungsphase evaluiert und dann zur Wiederverwendung in ein Repository aufgenommen. Zu jedem Zeitpunkt der Methodenmodellierung und -auswahl sollten Entwickler jeweils die Kontextspezifikationen und damit die Bedürfnisse der beteiligten Aktoren beachten. Dadurch wird eine weitere laufende

Evaluation impliziert, die zu verbesserten Ergebnissen und beschleunigten Konsensprozessen führt.

Das Informationsmodell in Tabelle 31 fasst die Spezifikation einer Methode zusammen.

Tabelle 31: Informationsmodell Methode

| Kategorie | Beschreibung |
|--------------------------------|---|
| <i>Allgemeine Daten</i> | |
| Dublin Core | Dublin Core-Elemente zur Methodenspezifikation |
| Referenz | Referenz auf weiteres Informationsmodell |
| <i>Beschreibung</i> | |
| Name | Name einer Methode |
| Quelle | Quelle der Methode |
| Template | Referenz auf Vorlage, Format der Vorlage |
| <i>Setting</i> | |
| Kontext | Kontextspezifikation |
| Raum | räumlicher Kontext |
| Zeit | zeitlicher Kontext |
| <i>Phasen</i> | |
| Phase | Name der Phase einer Methode |
| Art | Art der Phase |
| Anordnung | Beziehung zu Phase, Reihenfolge, Sequenzoperator (sequenziell, parallel, nebenläufig, frei) |
| Durchläufe | Anzahl der Durchläufe einer Phase |
| <i>Interaktion</i> | |
| Beschreibung | Beschreibung der Interaktion |
| Rolle | Rolle der Aktoren, Beschreibung |
| Typ | Typ der Interaktion |
| Topologie | unidirektional, bidirektional |
| Verlauf | synchron, asynchron |
| Anwendungen | Kommunikationsanwendungen |
| Referenz | Verweis auf weitere Kommunikationsspezifikation |
| <i>Präsentation</i> | |
| Typ | Typ eines Präsentationsobjektes |
| Anwendung | Anwendung zur Präsentation |
| Referenz | Verweis auf weitere Präsentationsspezifikationen |
| <i>Evaluation</i> | |
| Form | Art der Evaluation (Prüfung, Übungsszenario) |
| Anwendung | Referenz zu Evaluationsanwendung |
| Evaluation | Referenz auf Lernobjektevaluation |

Bei der Auswahl einer Methode kann der Lernende dann einerseits auf Hinweise zurückgreifen, andererseits können verschiedene Templates zur Implementierung einer Methode gestaltet werden. Templates umfassen Hinweise, Kommunikations-, Präsentations- und Evaluationsanwendungen, so dass nur noch nähere Erläuterungen für Lernende und Inhalte eingefügt werden müssen. Dadurch wird die Verwendung technisch anspruchsvoller Methoden für Lehrende wesentlich vereinfacht.

5.5.4 Lernumgebungsdesign

Das Design einer Lernumgebung fasst die drei vorhergehenden Aktivitätsbereiche zusammen. Es wird eine prototypische Lernumgebung generiert. Zunächst wird die Struktur eines Kurses festgelegt. Dabei werden Lernobjekte und Phasen einer Methode mit den entsprechenden Templates verknüpft. Es entsteht eine Lernumgebung, die Schlagwörter bezüglich der Inhalte, methodische Erläuterungen und Möglichkeiten der benutzerspezifischen Anpassung zur Verfügung stellt. Beispiele dieser Lernumgebungen werden in den Abschnitten 6.2.1 bis 6.2.3 vorgestellt.

Die umfangreiche Evaluation durch Lehrende, Lernende, Entwickler und Designer steht im Vordergrund. Inhaltliche, technische und didaktische Aspekte werden in dieser Aktivität analysiert und durch einen Kriterienkatalog bewertet. Die Änderungsaufgaben gehen in vorhergehende Aktivitäten ein und führen zu einer Konsensfeststellung. Abschließend erfolgt die Planung der Ebene ELM-E.

5.5.5 Zusammenfassung ELM-D

In ELM-D wird das Design einer Lernumgebung auf Kursebene erstellt. Dabei werden Inhalte recherchiert, bewertet und in einer SCORM-konformen Struktur (CSF) abgebildet. Weiterhin werden Metadaten zu einzelnen Lernobjekten spezifiziert und teilweise automatisch generiert. Es werden potenzielle Benutzergruppen beschrieben, um eine Basis zur Benutzeradaptation zu liefern.

Hervorzuheben ist insbesondere die Modellierung didaktischer Methoden, die den Nutzungskontext, eine Phasenstruktur sowie Empfehlungen zu Präsentation, Kommunikation und Evaluation bieten. Methoden können einerseits als Entscheidungs- und Designunterstützung dienen, andererseits können durch die Verwendung von Templates prototypische Lernumgebungen generiert werden, die spezifische Methoden implementieren. Dadurch werden die didaktischen Möglichkeiten von Lehrenden wie Entwicklern wesentlich erweitert. Die Methoden liegen zudem in einem Repository vor und können somit organisationsübergreifend genutzt und wiederverwendet werden.

Maßgebliche Ergebnisse dieser Ebene sind eine prototypische Lernumgebung, die Beschreibung von Lernobjekten, Aktoren, Methoden und verschiedene Metadatenpezifikationen. Diese können ebenfalls von anderen Systemen verwendet und bearbeitet werden.

5.6 Vorgehensmodell zur Entwicklung von Lerneinheiten (ELM-E)

In der Ebene ELM-E wird die Gestaltung einzelner Lerneinheiten vorgenommen. Die Repräsentation der Inhalte wird durch die Verwendung der *Learning Material Markup Language (LMML)* unterstützt (siehe 4.4.5.2). Des Weiteren wird der genaue Ablauf einer Lernumgebung beschrieben. Jede Lerneinheit umfasst ergänzende Kommunikations-, Präsentations- und Evaluationskomponenten. Abschließend erfolgen Test, Bewertung und Implementierung der Lernumgebung, die unter Verwendung der ELM-Run-Time evaluiert und genutzt werden kann.

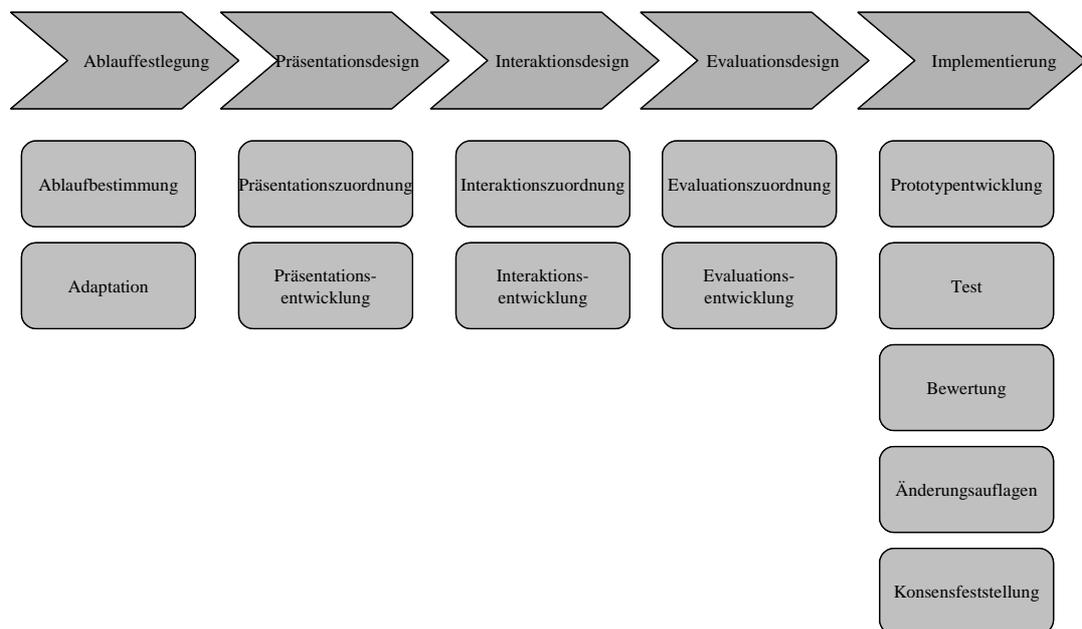


Abbildung 54: Wertschöpfungskette ELM-E

Die Aktivitäten und das Datenschema von ELM-E sind in Abbildung 54 und Abbildung 55 zusammengefasst. Die einzelnen Aktivitäten sind nicht in sequenzieller Reihenfolge zu bearbeiten, da die jeweiligen Gestaltungsentscheidungen einer einzelnen Aktivität auch in den anderen Aktivitäten berücksichtigt werden müssen. Daher wird hier eine nebenläufige Vorgehensweise vorgeschlagen.

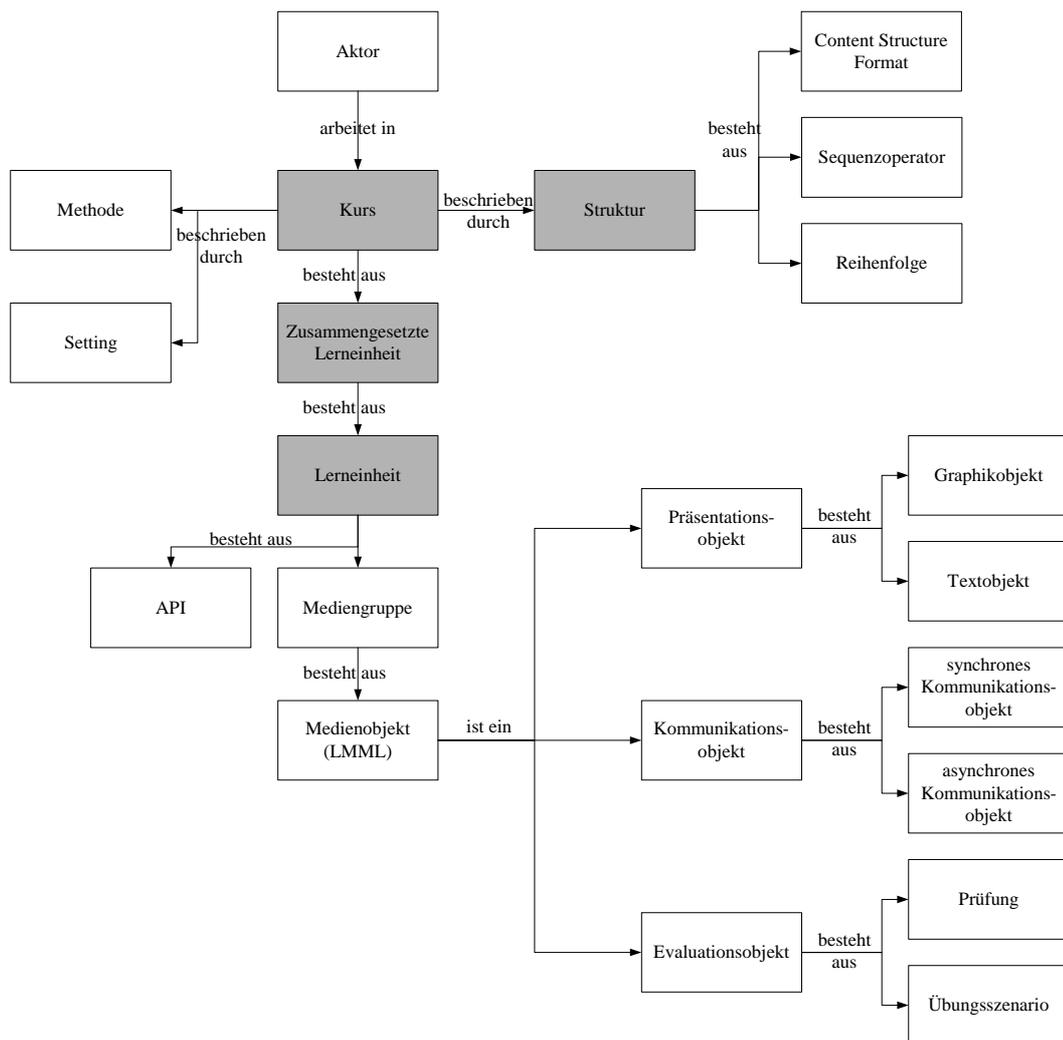


Abbildung 55: Datenschema Lerneinheit

5.6.1 ELM-E: Ablauffestlegung

In dieser Phase wird der in ELM-D bestimmte grobe *Ablauf* weiter detailliert. Dabei können die Empfehlungen zum Ablauf einer Methode verwendet werden, in der bereits Vorschläge spezifiziert sind, in welcher Reihenfolge und Sequenzierung Lernereinheiten bearbeitet werden können. Zudem werden die räumlichen und zeitlichen Faktoren genauer spezifiziert. Das bedeutet, dass ein Setting im Detail abgebildet wird. Dieses umfasst zum Beispiel die Definition von Präsenz- oder Fernlernphasen oder die Spezifikation synchroner und asynchroner Phasen. Weiterhin können die Empfehlungen der Methoden modifiziert werden.

Insbesondere müssen in dieser Phase die *sachlogischen* Zusammenhänge genauer spezifiziert werden, die auch als *Lernwege* bezeichnet werden. Bei einer *linearen* Abfolge müssen Lernereinheiten aufeinanderfolgend bearbeitet werden. Dieses wird insbesondere verwendet, wenn Lerninhalte aufeinander aufbauen oder eine Herleitung die sequenzielle Abfolge notwendig macht. Weiterhin kann die Abfolge *frei wählbar* sein. So kann in explorativen

Lernumgebungen der Lernende selbst bestimmen, welche Einheit bearbeitet werden soll. *Parallel* zu bearbeitende Lerneinheiten können bzw. müssen in einer Session bearbeitet werden. Dieses kann insbesondere in dem Fall sinnvoll sein, wenn verwandte Konzepte analysiert und verglichen werden sollen. Weiterhin muss spezifiziert werden, ob eine Abfolge zwingend eingehalten werden muss oder nur eine Empfehlung gegeben wird.

Im Rahmen einer *zeitlichen* Spezifikation können Zeitbegrenzungen vorgenommen werden. Als Beispiel seien Planspiele angeführt, die einen Wettbewerb zwischen einzelnen Teilnehmern einbeziehen. So muss in dieser Lernmethode eine zeitliche Begrenzung für die Erarbeitung eines Ergebnisses angegeben werden. Sollen keine Begrenzungen spezifiziert werden, so können zeitliche *Empfehlungen* vorgegeben werden, die für Lernende wie Lehrende als Richtlinie verwendet werden können. Wird ein konkreter Kurs geplant, so können in dieser Phase bereits *Zeitpunkte* für einzelne Lerneinheiten angegeben werden, die später vom Lernmanagementsystem in die Kursplanung einbezogen werden.

In dieser Phase kann zudem eine Benutzeradaptation vorgenommen werden, indem z. B. unterschiedliche Lernwege oder Lernzeiten für verschiedene Benutzergruppen definiert werden.

5.6.2 ELM-E: Präsentationsdesign

Der zweite Aktivitätsbereich umfasst die Erstellung der Lerninhalte und deren Präsentation. Während bisher nur Inhaltsstrukturen und Quellen erfasst wurden, werden nun konkrete Inhalte (Texte, Graphiken, Videos, Animationen etc.) erstellt. Hierfür wird in ELM die Repräsentation durch LMML vorgeschlagen. Es können jedoch auch beliebige andere Formate einbezogen werden. Eine Lerneinheit wird in Form eines LMML-Dokumentes repräsentiert; sie besteht aus einer Mediengruppe, die wiederum zusammengehörige Medienobjekte zusammenfasst. Ein Medienobjekt entspricht dabei Assets aus SCORM.

LMML-Dokumente gliedern sich in *Sections*, wobei der Inhalt semantisch gekennzeichnet ist. Dabei können folgende Typen zur Kennzeichnung einzelner Elemente verwendet werden:

Tabelle 32: LMML-Typen

| | | |
|--------------|----------------|----------------|
| • paragraph | • exercise | • task |
| • conclusion | • motivation | • solution |
| • definition | • remark | • bibliography |
| • example | • illustration | |

Jedes Element muss zusätzlich als Attribut einen Titel und einen Label besitzen. Die Elemente dürfen wiederum folgende child-Elemente haben:

- *text* enthält einen in HTML-formatierten Text und hat die Attribute *title* (Name des Text Objekts), *label* (Ident, dient als Sprungmarke) und *type* (MIME-Typ, Default: „text/html“).
- *image* enthält Graphiken und hat die Attribute *title* (Name des Image Objekts), *label* (Ident, dient als Sprungmarke), *align* (dient zur Formatierung) und *uri* (Unified Resource Identifier des Bildes).
- *sound* beinhaltet Audioformate und hat die Attribute *title*, *label* und *uri*.
- *animation* enthält Animationen und hat die Attribute *title*, *label* und *uri*.
- *referncesLink* beschreibt einen internen Verweis und hat die Attribute *target* (Name des Ankers im Zieldokument) und *target_modul* (Name des Zieldokuments).
- *externalLink* ist ein externer Verweis und enthält die Attribute *type* (Art des Links (link/html)) und *uri* (Ort des Zieldokuments).
- *citesLink* ist ein Link zu einer Zitation und hat die attribute *bibkey* (Name des Ankers im bibliography Modul) und *bib_modul* (enthält den Namen des bibliography Moduls).

Durch dieses Format können Lerninhalte mit semantischen Informationen repräsentiert werden. Um die Präsentation den Benutzerpräferenzen anzupassen (siehe 5.5.2), müssen Style Sheets zur Beschreibung der Präsentation entwickelt werden. Ein Beispiel wird dazu in Abschnitt 6.2 beschrieben. Weiterhin können externe Autorensysteme zur Erstellung spezifischer Lerninhalte eingebunden werden. ELM verwendet derzeit einen textorientierten Editor zur Bearbeitung von LMML-Dokumenten. Eine Erweiterung zu spezifischen Autorensystemen oder gängigen XML-Entwicklungsumgebungen ist vorgesehen, um die Präferenzen der Entwickler und Autoren zu berücksichtigen.

Hinzu kommt die Verfügbarkeit einer API, die den Aufruf und die Steuerung von Lerneinheiten durch ein LMS ermöglicht. Derzeit können Autoren, die SCORM-konforme Lerneinheiten entwickeln, auf zwei Datenmodelle zugreifen:

- CMI spezifiziert Daten, die zwischen LMS und Lerneinheit übergeben werden (siehe 4.4.6.1).
- Das ELM-Aktorenmodell ermöglicht den Austausch von Aktorendaten.

Diese können von Autoren zur Anpassung von Lerneinheiten verwendet werden. In ELM muss spezifiziert werden, ob eine Entwicklung SCORM-konform generiert werden soll. Das hat den Vorteil, dass die generierten Lerneinheiten in verschiedenen LMS verwendet werden können. Ist diese Option gewählt, müssen allerdings auch extern eingebundene Lerneinheiten so verändert werden, dass ihnen eine entsprechende API zugeordnet ist. Die Anpassungsfunktionen der Lerneinheiten müssen dann ebenfalls die Verwendung der API sicherstellen.

5.6.3 ELM-E: Interaktionsdesign

Unter Verwendung der Methodenempfehlung erfolgt die Ausgestaltung der *Kommunikation*. Dabei werden Kommunikationsanwendungen empfohlen, die für das jeweilige Setting (räumlich und zeitlich) geeignet sind. Derzeit werden folgende Anwendungen unterstützt:

- *Synchrone Kommunikation*: Videokonferenz, Chat, Application Sharing, Instant Messaging
- *Asynchrone Kommunikation*: Audio-Streaming, Video-Streaming, Forum, Newsgroup, E-Mail, Annotationen, Mehrbenutzereditoren

Die genannten Typen sind jeweils mit einer Applikation, die in der Kontextanalyse erfasst wurde, assoziiert und können somit eingebunden werden. Ebenso können zur Generierung von Übungsszenarien externe Applikationen (z. B. betriebswirtschaftliche Anwendungssysteme) berücksichtigt werden.

Die Verwendung generischer Kommunikationsobjekte bietet die Möglichkeit, projektspezifische Anwendungen zu definieren, die bereits in einer Organisation vorhanden sind. Diese Anwendungen können in einer Lernumgebung verwendet werden, so dass keine neuen Applikationen entwickelt oder beschafft werden müssen. Zudem können Lernende dadurch im Lernprozess auf gewohnte Anwendungen zurückgreifen und müssen keine neuen Applikationen erlernen.

5.6.4 ELM-E: Evaluationsdesign

In der Evaluationsphase wird spezifiziert, welche Mechanismen zur Evaluation einer Lernumgebung zur Verfügung gestellt werden sollen. Dadurch ist bereits in der Planung einer Lernumgebung die Qualitätssicherung integraler Bestandteil des Entwicklungsprozesses. Es kann spezifiziert werden, welche Evaluationstypen zu welchem Zeitpunkt verfügbar sein sollen, um Feedback über eine Lernumgebung zu erhalten und diese Information mit nachfolgenden Autoren, Lernenden und Lehrenden zu teilen.

Des Weiteren bezieht sich die Phase auf die Lernerfolgsüberprüfung bzw. die Gestaltung von Übungen und Prüfungen. In vielen Fällen ist die Prüfung nicht Bestandteil der Lernumgebung, da Prüfungen aufgrund des Authentifizierungsproblems häufig als Präsenzveranstaltung durchgeführt werden. Dennoch sollten verschiedene Evaluationstypen als Unterstützungsfunktion für Lehrende und Lernende zur Einschätzung der Lernerfolgs zur Verfügung stehen. Daher stehen derzeit folgende Evaluationsmechanismen in ELM zur Verfügung:

- *Fragebögen*: Offene Fragen, Multiple Choice, Zuordnung
- *Interviewverfahren*: Nutzung synchroner Kommunikationsobjekte
- *Beobachtungstechniken*: Beobachtung von Tracking-Informationen
- *Testverfahren*: Prüfung und Bewertung
- *Übungsszenarien*: Einbindung externer Applikationen (verfügbar als Methode)
- *Verweise*: Verweis auf eine Präsenzveranstaltung, in der eine Prüfung erfolgt

Derzeit werden diese Mechanismen den Lernenden lediglich im Sinne einer persönlichen Erfolgskontrolle zur Verfügung gestellt. Aufgrund rechtlicher Fragestellungen werden Prüfungen derzeit in Form traditioneller Präsenzprüfungen durchgeführt. Weiterhin stehen Übungsszenarien zur Verfügung, die zwar als Methode definiert sind, jedoch Evaluationsaktivitäten umfassen. Zusammenfassend beschreibt Tabelle 33 die gesamte Spezifikation einer Lerneinheit:

Tabelle 33: Informationsmodell Lerneinheit

| Kategorie | Beschreibung |
|--------------------------------|--|
| <i>Allgemeine Daten</i> | |
| Dublin Core | Dublin Core-Elemente für Lerneinheit |
| Referenz | Referenz auf weiteres Informationsmodell |
| <i>Beschreibung</i> | |
| LOM | Beschreibung nach LOM |
| SCO | SCO-Daten |
| Kontext | Verweis auf Kontextmodell |
| Methode | Verweis auf Methodenmodell |
| <i>Parameter</i> | |
| CMI | Parameter zur Übergabe: CMI |
| Aktorenmodell | Parameter zur Übergabe: ELM-Aktorenmodell |
| Weitere Parameter | weitere Parameter, Datenmodell der Parameter |
| API | API |
| <i>Interaktion</i> | |
| Beschreibung | Beschreibung der Interaktion |
| Rolle | Rolle der Aktoren, Beschreibung |
| Typ | Typ der Interaktion |
| Topologie | unidirektional, bidirektional |
| Verlauf | synchron, asynchron |
| Anwendungen | Kommunikationsanwendungen |
| Referenz | Verweis auf weitere Kommunikationsspezifikation |
| <i>Präsentation</i> | |
| Typ | Typ eines Präsentationsobjektes |
| Anwendung | Anwendung zur Präsentation |
| Referenz | Verweis auf weitere Präsentationsspezifikationen |
| <i>Evaluation</i> | |
| Form | Art der Evaluation (Prüfung, Übungsszenario) |
| Anwendung | Referenz zu Evaluationsanwendung |
| Evaluation | Referenz auf Lernobjektevaluation |

Nach der Spezifikation von Ablauf, Kommunikation, Präsentation und Evaluation erfolgt die Erstellung eines weiteren *Prototypen*, der als Basis einer Abschlussevaluation dient.

Dieser wird sowohl vom Projektteam als auch von einer Testgruppe evaluiert und bewertet. Die Änderungsaufgaben werden dann eingearbeitet, bis ein Konsens zum Projektabschluss führt.

5.6.5 ELM-Run-Time

ELM-Run-Time (ELM-RT) bietet die Basisfunktionen eines SCORM-konformen LMS und kann somit als Testumgebung für SCORM-konforme Lernumgebungen verstanden werden. ELM-RT unterstützt die prototypische Entwicklung von Lernumgebungen mit dem Ziel der Evaluation derselbigen. Mit ELM entwickelte Lernumgebungen können jedoch auch in anderen Systemen, wie zum Beispiel Learning Space, Clear Campus oder WebCT, verwendet werden. Sobald SCORM in den marktgängigen LMS verfügbar ist, sind auch andere Testumgebungen möglich. Da die derzeit geläufigen Systeme nicht alle Funktionen enthalten, die von ELM zur Verfügung gestellt werden, wurde mit ELM-Run-Time ein eigenständiges Testsystem entwickelt.

Zunächst werden die Basisfunktionen von ELM-RT erläutert. Dabei werden die Interdependenzen zum Entwicklungssystem beschrieben. Die in hellgrauer Farbe gekennzeichneten Funktionen sind für Lernende maßgeblich, die dunkelgrau gekennzeichneten sind für Lehrende bzw. Administratoren von besonderer Bedeutung (Abbildung 56).

Der Aktivitätsbereich *Anmeldung* umfasst einerseits die Anmeldung von Aktoren an das System. Zunächst werden Benutzerprofile angelegt und entsprechende initiale, rollenabhängige Berechtigungen vergeben. Dabei werden Stammdaten der Benutzer angelegt, die den persönlichen Daten aus PAPI (siehe 5.5.2) entsprechen. Wurden in ELM bereits Daten über die Aktoren erfasst (während der Teamzusammensetzung, 5.4.2, und der Benutzeranalyse, 5.5.2), so können diese übernommen werden. Die Kursadaptation bietet die Möglichkeit Kurse aufgrund eines Benutzerprofils anzupassen.

Der Aktivitätsbereich *Portfolioverwaltung* umfasst die administrativen Funktionen, die in einem Lernprozess benötigt werden. Die Nutzerverwaltung bietet die Möglichkeit, Daten der teilnehmenden Aktoren zu verwalten und zu verändern. Des Weiteren muss eine Prüfungsverwaltung zur Verfügung gestellt werden. Eine Prüfung bezieht sich auf einen Kurs und wird den entsprechenden Lernenden und den Lehrenden zugeordnet.

Aus Sicht der Lernenden sind die Funktionen der Kursauswahl und der übergeordneten Programmauswahl (als Zusammenfassung mehrerer Kurse) maßgeblich. Lernende können durch diese Funktion Programme und entsprechende Kurse belegen bzw. abmelden. Wird ein Kurs und die korrespondierende Prüfung erfolgreich absolviert, so werden die entsprechenden Kompetenzen zum Profil hinzugefügt. Für Lehrende bzw. Autoren besteht die Funktion Kursauswahl in der Zusammenstellung von Kursen bzw. Programmen. Aus ELM bzw. aus einem Repository werden Kurse hinzugefügt oder neu erstellt. Die Entwicklung wird dabei jedoch mithilfe der ELM-Applikation durchgeführt. ELM-Run-Time ist SCORM-konform, daher wird aus der Entwicklungsumgebung entweder die Kursstruktur unter Verwendung des Content Structure Format oder durch die Verwendung von Content Packages exportiert.

Der Lernprozess erfolgt in den Aktivitätsbereichen *Aktivität* und *Betreuung*. Diese Bereiche sind im Prozessmodell unterteilt, da häufig unterschiedliche Auswertungen für diese Bereiche (z. B. aus Sicht der Kostenrechnung) erstellt werden, dennoch sind sie nicht getrennt voneinander zu betrachten. Aus Sicht des Lernenden werden Lerninhalte bearbeitet und in Übungen angewendet. Hinzu kommt die Kommunikation mit anderen Akteuren. Die einzelnen Aktivitäten werden durch eine Tracking-Funktion erfasst, um Auswertungen über Lernverhalten als Basis zur individuellen Anpassung zu erhalten. Die Betreuung besteht aus einer Interaktion zwischen Lehrenden und Lernenden, der Problemlösung und entsprechender Nachbearbeitung.

Der Bereich der *Prüfung* besteht aus den Funktionen der Anmeldung, Durchführung, Bewertung und anschließender Zertifizierung. Hinzu kommt die Funktion des Feedbacks, die eine Bewertung der Lernumgebung ermöglicht. Gerade diese Funktion ist auch im Entwicklungsprozess von besonderer Bedeutung, da hier die einzelnen Prototypen evaluiert werden.

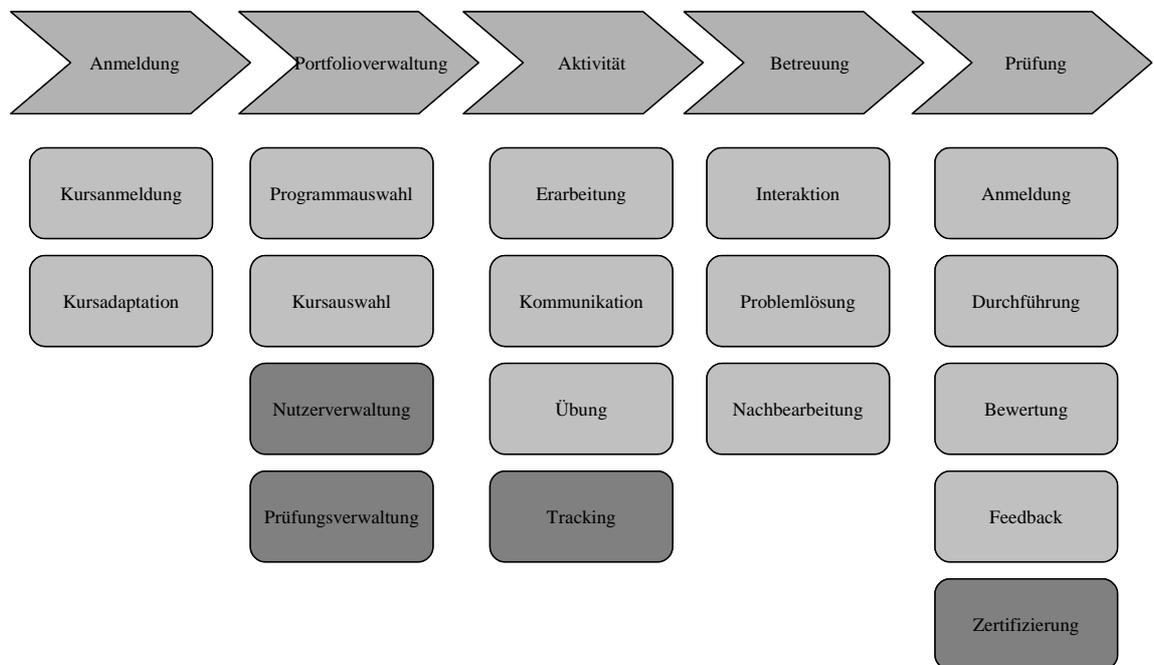


Abbildung 56: ELM-Run-Time

ELM-Run-Time enthält somit die Basisfunktionen eines Lernmanagementsystems und ermöglicht somit eine Evaluation von Lernumgebungen in einem realen System. Insbesondere wird das System als Testumgebung für die Erprobung von Standards in einem Lernmanagementsystem verwendet.

6 Umsetzung des Essener-Lern-Modells

In diesem Kapitel wird die Umsetzung des in Kapitel 5 konzeptionell erläuterten Essener-Lern-Modells vorgestellt. Zunächst wird die ELM-Applikation beschrieben. Diese Applikation basiert auf dem Essener-Lern-Modell und ist ein unterstützendes Werkzeug für Konzeption, Design und Implementierung von Lernumgebungen. Zunächst wird die Implementierung kurz vorgestellt, um die Einsetzbarkeit und den Aufbau der Applikation darzustellen. In ausgewählten Anwendungsbeispielen wird verdeutlicht, wie Entwicklungsprozesse von der ELM-Applikation unterstützt werden.

6.1 Beschreibung der ELM-Applikation

Die ELM-Applikation ist das Werkzeug, das auf Basis des Essener-Lern-Modells entwickelt wurde [vgl. Pawl2001]. In diesem Abschnitt werden zunächst Entwicklungsrichtlinien und –entscheidungen vorgestellt. Dabei werden die Architektur und die einzelnen Komponenten der ELM-Applikation beschrieben. Daran anschließend werden die Funktionen des Modells im Detail erläutert. Aus Gründen der Übersichtlichkeit werden Datenmodelle nur dann weitergehend betrachtet, wenn es zum Verständnis der Arbeit notwendig ist.

6.1.1 Entwicklungsentscheidungen

Im Folgenden werden die Entwicklungsrichtlinien für die ELM-Applikation beschrieben, die auf den Kriterien aus Abschnitt 5.1 aufbauen. Hinzu kommt die Beschreibung der technischen Architektur, wobei jeweils Bezug auf die konzeptionelle Architektur aus Abschnitt 5.3 genommen wird.

Die Auswahl der Systemarchitektur, der Entwicklungsumgebung und der Programmiersprache erfolgte unter Berücksichtigung der in Abschnitt 5.1 beschriebenen Kriterien der Softwareentwicklung. Ein Fokus wurde auf die Aspekte der Portabilität und Interoperabilität gelegt, da nicht nur ELM-Lernumgebungen, sondern auch die Entwicklungsumgebung systemunabhängig verfügbar sein muss.

6.1.1.1 Programmiersprache

Zur Entwicklung der ELM-Applikation wurde die Programmiersprache Java ausgewählt. Durch die Verwendung von Java ist die ELM-Applikation *interoperabel*. Weiterhin sind die Komponenten der Applikation wiederverwendbar und stehen somit z. B. für Open-Source-Projekte zur Verfügung. Ein weiterer maßgeblicher Faktor ist die *Benutzerfreundlichkeit*. Für die Programmiersprache Java liegen bereits zahlreiche Standardklassen zur Programmierung graphischer Benutzeroberflächen vor. Ebenso können bereits Standardklassen für den Datenbankzugriff oder die Dateiverwaltung verwendet werden. Ein weite-

res Argument für die Verwendung von Java ist die weite Verbreitung der Programmiersprache. Es ist davon auszugehen, dass Java oder entsprechende Weiterentwicklungen in den nächsten Jahren weiterhin existieren werden und somit eine langfristige Verwendbarkeit sichergestellt ist.

Als Entwicklungsumgebung wurde JBuilder 5.0 [Borl2001] verwendet. Diese Entwicklungsumgebung ermöglicht neben einer einfach zu bedienenden graphischen Benutzeroberfläche verschiedenste Dokumentationsfunktionen. So wird zum Beispiel eine webbasierte Dokumentation generiert, die gerade in verteilt arbeitenden Programmerteams genutzt werden kann.

6.1.1.2 Technische Architektur

Die nächste Entscheidung bezieht sich auf die technische Architektur. Dabei sind insbesondere Einsatzszenarien zu betrachten, in denen die Applikation verwendet wird:

Die Entwicklung von Lernumgebungen kann insbesondere durch die Faktoren Kooperationsgrad und Wiederverwendbarkeit beschrieben werden. Die Entwicklung erfolgt in Entwicklungsteams mit einem sehr hohen *Kooperationsgrad*. Das bedeutet, dass die Entwicklung sowohl räumlich als auch zeitlich verteilt erfolgt und die Kooperation und Kommunikation via Internet oder Intranet erfolgt. Dieses Szenario erfordert eine webbasierte Lösung. Des Weiteren sollen Lernumgebungen, Teilprodukte, wie auch einzelne Lern- oder Medienobjekte *wiederverwendbar* sein. Aus diesem Grund wird eine datenbankbasierte Lösung bevorzugt, um Speicherung, Suche und Retrieval einfach zu gestalten.

Die technische Architektur ist in Abbildung 57 zusammengefasst.

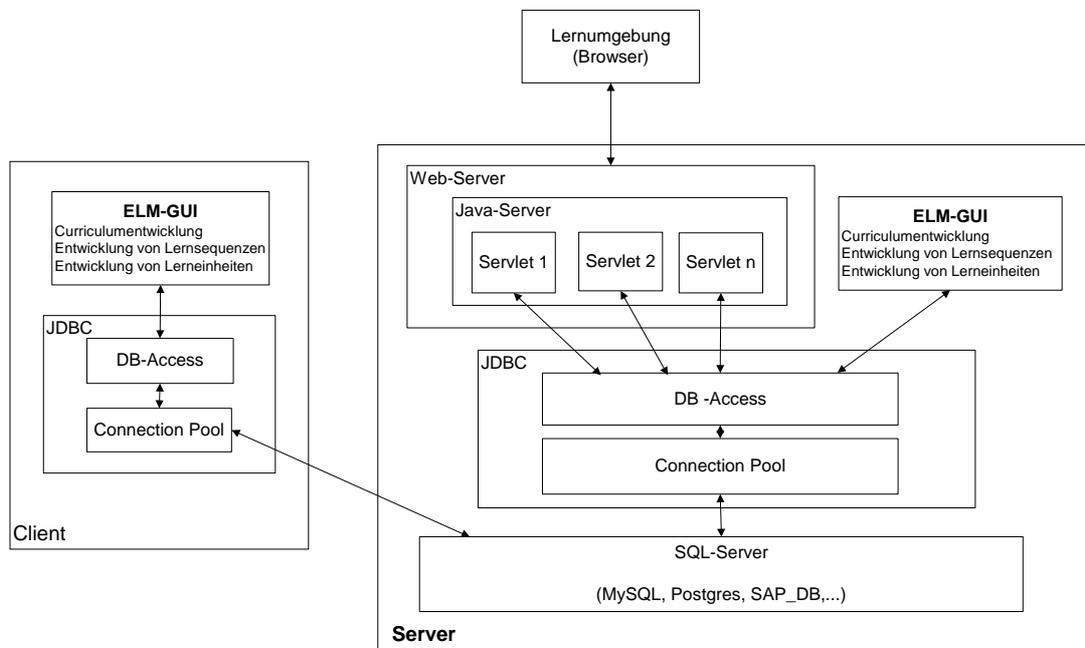


Abbildung 57: Technische Architektur der ELM-Applikation

Die technische Architektur lässt sich generell in drei Komponenten unterteilen:

- Die *Serveranwendung* stellt die zentralen Funktionen der Applikation zur Verfügung. Diese Komponente ist für die zentrale Verwaltung der Datenbank, die Benutzerverwaltung und die Verwaltung der Lernumgebungen verantwortlich. Sie basiert auf einem SQL-Server, wobei derzeit Versionen für die Verwendung von MySQL, SAP_DB und Postgres zur Verfügung stehen. Die Anbindung an den Datenbank-Server erfolgt mittels der Module *Connection Pool* und *DB-Access*. Diese ermöglichen den Datenbankzugriff und verwalten die entsprechenden Autorisierungs- und Zugriffsrechte.
- Das *ELM-GUI* (Graphical User Interface, Graphische Benutzeroberfläche) ist die Anwendungsumgebung, in der die Benutzer arbeiten. Auf diese Umgebung kann einerseits über den Server zugegriffen werden, andererseits kann über einzelne Clients gearbeitet werden. Für die zweite Alternative ist keine dauerhafte Internetanbindung notwendig, d. h., die Bearbeitung kann offline erfolgen. Die Synchronisation der Inhalte erfolgt über die Serveranwendung. Die Anwendungsumgebung wird im Detail im Abschnitt 6.1.2 beschrieben; an dieser Stelle soll nur eine Gesamtübersicht über die technische Architektur erfolgen.
- *Lernumgebungen* werden von der ELM-Applikation in XML generiert. Da die derzeit gängigen Browser (z. B. Internet Explorer 5.0, Netscape 4.6) die Verwendung von XSL-Stylesheets noch nicht vollständig unterstützen, werden XML- und XSL-Dateien serverseitig durch die XALAN-Applikation [Xala2001] in HTML-Dateien umgewandelt und sind dadurch browserunabhängig nutzbar. Die detaillierte Beschreibung der erzeugten Lernumgebungen und deren Aufbau erfolgt in Abschnitt 6.2.

6.1.2 Applikationen

Im folgenden Abschnitt werden kurz die Teilapplikationen der ELM-Applikation beschrieben. Dabei wurden für einzelne Teilbereiche des Entwicklungsprozesses kleinere und einfach zu handhabende Applikationen generiert. Diese können von Entwicklern genutzt werden, die nicht die gesamte Applikation nutzen, sondern nur Teilaufgaben erfüllen. Zusätzlich zum Gesamtsystem *ELM-All* sind folgende Applikationen verfügbar:

- ELM-LOM (ELM-Learning Object Metadata)
- ELM-User
- ELM-LOB (ELM-Learning Objectives)
- ELM-Editor
- ELM-Method
- ELM-Catalog

Die Applikation *ELM-All* umfasst sämtliche Funktionalitäten von ELM und wird in den folgenden Abschnitten im Detail erläutert. Dabei werden insbesondere die Umsetzungen der Ebenen ELM-C, ELM-D und ELM-E betrachtet.

Die Applikation *ELM-LOM* (ELM-Learning Object Metadata) ist eine Teilapplikation zur Bearbeitung von Lernobjekten (Abbildung 58). Dabei werden alle in ELM verfügbaren Funktionen abgedeckt, die bei der Bearbeitung von Lernobjekten benötigt werden. Dieses umfasst die Bearbeitung der Metadaten, Quellenerfassung, Teamzuordnung, Bildung und Bearbeitung von Lerneinheiten, Methodenzuordnung, Generieren von Prototypen und die Erstellung einer Evaluation. Durch diese vereinfachte Bearbeitung von Lerneinheiten werden insbesondere Entwickler unterstützt, die die Strukturierung und Bearbeitung von Lernobjekten vornehmen, wie zum Beispiel Content Provider oder Designer.

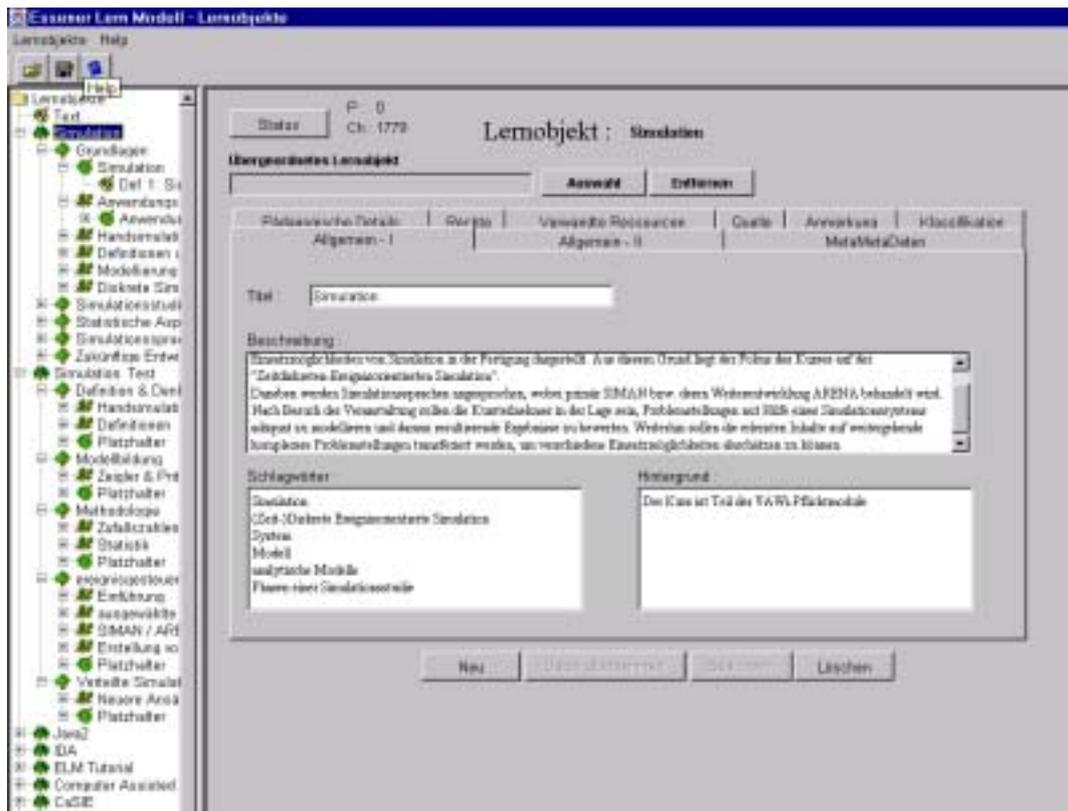


Abbildung 58: Teilapplikation ELM-LOM

Die Applikation *ELM-User* ist eine Applikation zur Bearbeitung der Daten von Lernenden und potenziellen Lerngruppen (Abbildung 59). Diese Teilapplikation wird insbesondere für das Lehrpersonal, aber z. B. auch Personalentwickler benötigt, da hier Daten über Kompetenzen, Vorwissen und Lernpräferenzen erfasst werden können.

Die Applikation *ELM-Method* wird zur Bearbeitung der Methodenbasis verwendet und wurde insbesondere für Lehrpersonal und didaktische Designer (Abbildung 61) entworfen. In dieser Applikation erfolgt die Modellierung didaktischer Methoden und das Erfassen von Erfahrungen mit den verfügbaren Methoden. Da der Bereich des didaktischen Designs ebenfalls häufig getrennt von der technologischen Konzeption durchgeführt wird, steht auch hier eine Teilapplikation zur Verfügung.

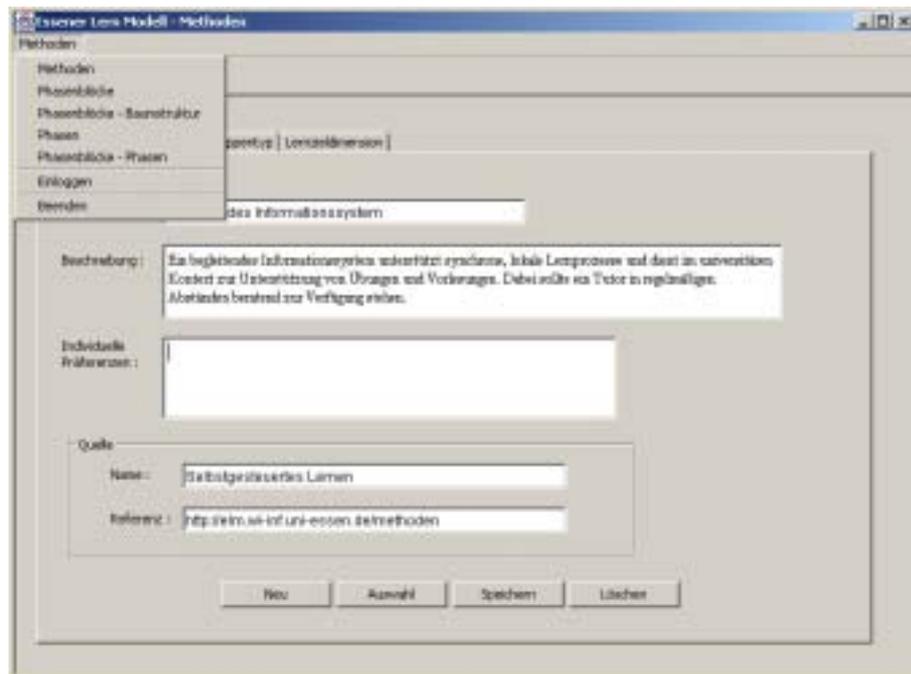


Abbildung 61: Teilapplikation ELM-Method

ELM-Catalog ist eine Applikation zur Bearbeitung von Katalogen für Metadaten. Wie bereits in Abschnitt 4.4.3 erläutert, ist die Verfügbarkeit und Pflege von Katalogen für die Ausprägungen der Metadatenattribute unerlässlich. So müssen zum Beispiel Curricula, Kompetenzen und inhaltliche Themen in Form von Thesauri oder Ontologien modelliert werden, um Suche, Retrieval und daraus folgend eine Wiederverwendung zu ermöglichen.

6.1.2.1 Anpassung

In diesem Abschnitt werden die Funktionen der ELM-Applikation zur projekt- und organisationsspezifischen Anpassung erläutert. Die allgemeinen Anpassungsfunktionen und Projektmanagementfunktionen stehen im Menü *Projektmanagement* zur Verfügung.

Wie bereits in der konzeptionellen Beschreibung erläutert (siehe Kapitel 5), wird das generische Essener-Lern-Modell auf die Bedürfnisse eines Projekts abgestimmt. Der erste Anpassungsschritt ist die Analyse und die Auswahl der benötigten Prozesse und deren Ver-

knüpfungen. Folgende Anpassungsfunktionen stehen im Menü *Projektmanagement*→*Listen* zur Verfügung:

- Auswahl, Anlegen und Gruppierung von Prozessen
- Spezifikation der Ablaufsteuerung
- Anpassung der Terminologie und Sprachauswahl
- Anpassung der Attribute der Metadaten
- Anpassung der verwendbaren MIME-Types

Die identifizierten Prozesse des Essener-Lern-Modells (siehe 5.4 bis 5.6) stehen in einer Auswahlliste im Menü *Projektmanagement*→*Listen*→*Prozesse* zur Verfügung und können projektspezifisch ausgewählt werden. Soll das Modell erweitert werden, können selbst definierte Prozesse an dieser Stelle eingefügt werden. Werden Prozesse aus Gründen der Übersichtlichkeit oder aus organisatorischen Gründen zusammengefasst, lassen sich in *Projektmanagement*→*Listen*→*Prozessgruppen* jeweils Gruppen von Prozessen bilden, die somit als übergeordnete Prozesse definiert werden.

Die Verknüpfung von Prozessen zur Ablaufsteuerung des Entwicklungsprozesses in ELM erfolgt im Menü *Projektmanagement*→*Listen*→*Prozessanforderungen*. In diesem Menü werden hierarchische Bedingungen oder Reihenfolgebedingungen spezifiziert. So ist es möglich, bestimmte Prozesse erst nach Beendigung anderer Prozesse zur Bearbeitung freizugeben. Abbildung 62 zeigt die Zuordnung von Prozessen und die Spezifikation von Freigabevoraussetzungen, während die Zuordnung zu Prozessgruppen in Abbildung 63 skizziert ist.

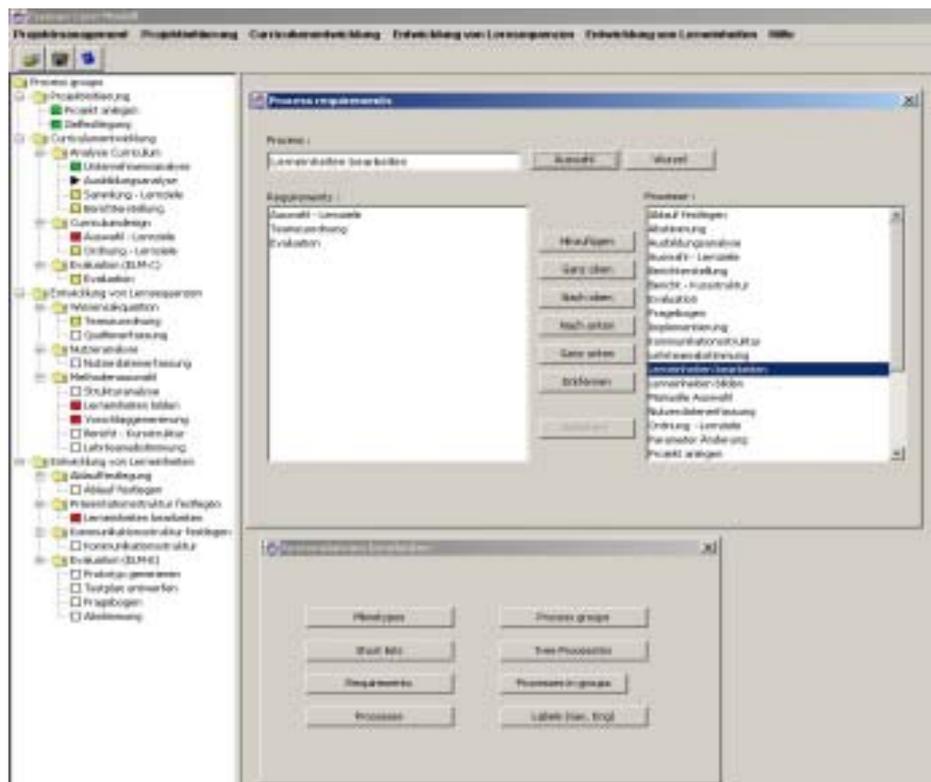


Abbildung 62: Projektmanagement Prozessanforderungen

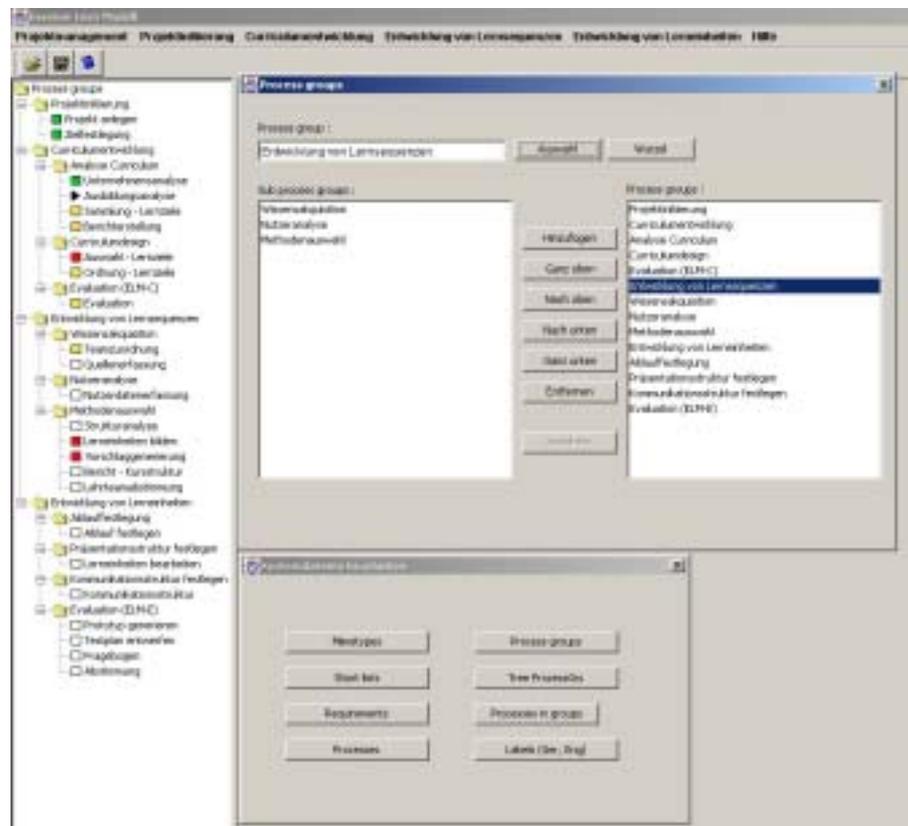


Abbildung 63: Projektmanagement Prozessgruppen

Neben der Anpassung der Prozesse ist ebenfalls die Anpassung der Terminologie möglich. Derzeit steht eine Standardterminologie zur Verfügung, wobei im Menü *Projektmanagement*→*Listen*→*Labels* zwischen deutschen und englischen Bezeichnungen ausgewählt werden kann.

Insbesondere bei der Bearbeitung von Metadaten ist die Bearbeitung der möglichen Ausprägungen der Attribute von besonderer Bedeutung. So können organisationsspezifische Ausprägungen (z. B. bei der Spezifikation von Katalogen) im Menü *Projektmanagement*→*Listen*→*Short Lists* bearbeitet werden (siehe Abbildung 64). Derzeit unterscheidet ELM zwischen kognitiven, affektiven, psychomotorischen sowie sozialen Lernzielen. Werden zusätzliche oder abweichende Lernziele verwendet, so können diese in diesem Menüpunkt bearbeitet und spezifiziert werden.

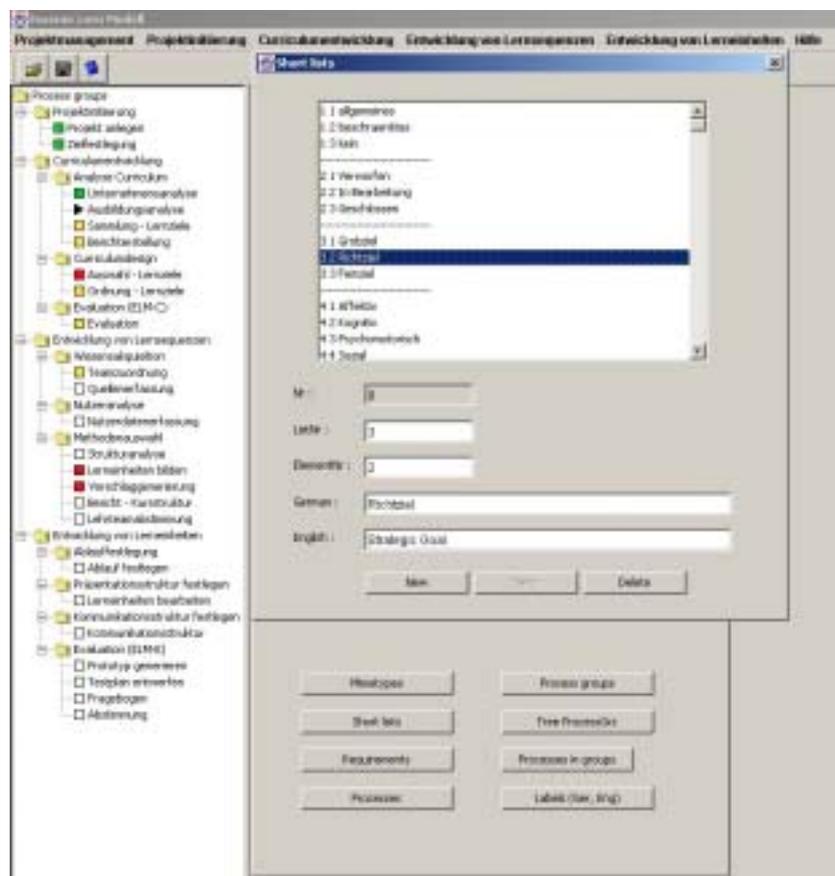


Abbildung 64: Projektmanagement Listen

Des Weiteren werden im Menü *Projektmanagement*→*Listen*→*Mimetypes* diejenigen Applikationstypen spezifiziert, die im nachfolgenden Entwicklungsprozess verwendet werden können. Die Veränderung der Prozesse und der Terminologie kann auch während des Entwicklungsprozesses erfolgen. Mögliche Inkonsistenzen werden dabei erkannt und können nachträglich aufgelöst werden. Dennoch sollte generell die Anpassung mit Beginn des Einsatzes von ELM durchgeführt werden.

6.1.2.2 ELM-C

Als maßgebliche Ergebnisse werden in der Phase ELM-C folgende Spezifikationen und Produkte entwickelt (siehe auch Abschnitt 5.4):

- Projektziele und Projektplanung
- Kontextbeschreibung durch Spezifikation der Organisationsanalyse, IT-Analyse und Ausbildungsanalyse
- Curriculum durch die Spezifikation von Lernzielen
- Kursstruktur durch Spezifikation von Lernobjekten
- Prototyp der Lernumgebung (Navigation, inhaltliche Gliederung, Lernziele)
- Evaluation durch Konsensfeststellung

Projektinitiierung

Im Modul ELM-C erfolgt zunächst die Initiierung eines Projekts. Dabei wird eine Beschreibung des Gesamtprojekts angefertigt. Diese ist nicht als formale Abgrenzung im Sinne eines Pflichtenheftes zu sehen, sondern soll zum besseren Verständnis dienen (siehe Abbildung 65).



Abbildung 65: ELM-C Projektbeschreibung

Hinzu kommt die Bestimmung von Projektzielen. Sie dienen als Basis für Evaluationen während und nach Beendigung des Projekts. Daher sollten die Ziele nicht einmalig zu Beginn des Projekts festgelegt, sondern laufend angepasst werden und somit den jeweiligen Konsens der Akteure widerspiegeln. Bereits bestehende Ziele können mittels des Buttons *Auswahl* bearbeitet und mit *Neu* hinzugefügt werden. Die Projektziele liegen in einer Auswahlliste vor und können somit wiederverwendet werden. Das bedeutet, dass Projektziele in weiteren Projekten übernommen werden können und nur teilweise überarbeitet werden müssen. Dies ist insbesondere bei Zielen sinnvoll, die die Einhaltung von organisationspezifischen Standards und Richtlinien betreffen (z. B. Richtlinien zur Corporate Identity).

Des Weiteren werden in dieser Initiierungsphase sowohl die Verantwortlichkeiten als auch eine Organisationsstruktur innerhalb des Projekts abgebildet. Generell können Aufgaben von Abteilungen, Teams oder Personen bearbeitet werden. Zu Beginn des Projekts werden in der Regel zunächst Teams und zugehörige Personen festgelegt, die jedoch im weiteren Verlauf verändert und ergänzt werden. Gerade inhaltliche Schwerpunkte werden erst während der Entwicklung bestimmt und sind zu Beginn noch nicht im Detail abzusehen. So kann eine zuständige Abteilung spezifiziert, die entsprechende Experten zu einem späteren Zeitpunkt benennt (z. B. Domänenexperten). Somit wird bereits zu diesem Zeitpunkt die Zuständigkeit festgelegt, während die personelle Zuordnung erst später erfolgt.

Neben der Planung der personellen Ressourcen erfolgt eine initiale Zeitplanung. Dabei werden die Prozessgruppen und Prozesse erfasst, die bei der Anpassung ausgewählt wurden. Das Menü *Projektmanagement* → *Zeitmanagement* umfasst eine graphische Übersicht über die zeitliche und personelle Planung des Projekts (siehe Abbildung 66).



Abbildung 66: ELM-C Zeitmanagement

Neben der Erfassung der Prozesse wird der Status angezeigt: *Grün* bedeutet, dass ein Prozess durch Konsensfeststellung abgeschlossen ist; *gelb* bedeutet Bearbeitung bzw. Überarbeitung; *rot* bedeutet, dass ein Prozess noch nicht begonnen wurde. Des Weiteren erfolgt eine Empfehlung von Prozessen, die auf der festgelegten Prozessreihenfolge basiert. Abweichungen von dieser Reihenfolge sind zwar möglich, werden jedoch durch eine rote Markierung als *nicht empfohlen* gekennzeichnet.

Kontextanalyse

In der Kontextanalyse wird der Kontext einer Organisation erfasst (siehe Abbildung 67). Zunächst werden im Menü *Curriculumentwicklung*→*Analyse Curriculum*→*Unternehmensanalyse* Basisdaten über die durchführende Organisation erfasst. Dabei werden die generellen Rahmenbedingungen einer Entwicklung festgelegt, indem die Organisation, deren Prozesse, Organisationsziele, -philosophie, -standards und Produkte bzw. Services beschrieben werden.

Darauf folgt die Analyse der IT-Struktur, also der bestehenden und nutzbaren Hardware, Software und Netzwerke. Diese werden jeweils in einer Auswahlliste erfasst und stehen im weiteren Verlauf zur Verfügung. Somit können Applikationen, die in einer Organisation bereit stehen, in die Entwicklung einer Lernumgebung integriert werden. Des Weiteren können diese Daten zur Orientierung dienen, welche Systeme und damit auch Benutzeroberflächen als bekannt vorauszusetzen sind.

The screenshot shows the 'Unternehmensanalyse' form in the 'Essener Lern-Modell' software. The form is titled 'Unternehmensanalyse' and has a 'Status' tab. It contains several input fields: 'Name' (University of Essen), 'Branche' (Academy), 'Adresse' (University of Applied Sciences Essen), 'Internet Zugang' (General), and 'Bandbreite' (100 Mbit/s). There are also two 'Bemerkung' (Remarks) text areas. A 'Speichern' (Save) button is at the bottom.

Abbildung 67: ELM-C Unternehmensanalyse

Im Menü *Curriculumentwicklung*→ *Analyse Curriculum*→*Ausbildungsanalyse* werden daraufhin existierende Bildungsmaßnahmen und korrespondierende Kompetenzen erfasst. Existierende Bildungsmaßnahmen können das Aggregationsniveau eines einzelnen Kurses bis hin zu vollständigen Personalentwicklungsprogrammen (z. B. Trainee-Programm, Management-Programm) haben. Jeder Bildungsmaßnahme werden Kompetenzen zugeordnet, die ein Lernender nach Abschluss der Maßnahme erwirbt und somit Lernzielen entsprechen. Die Kompetenzen lassen sich einzeln zu einer Auswahlliste hinzufügen und können als Netzwerk innerhalb einer Baumstruktur verknüpft werden (siehe Abbildung 68).

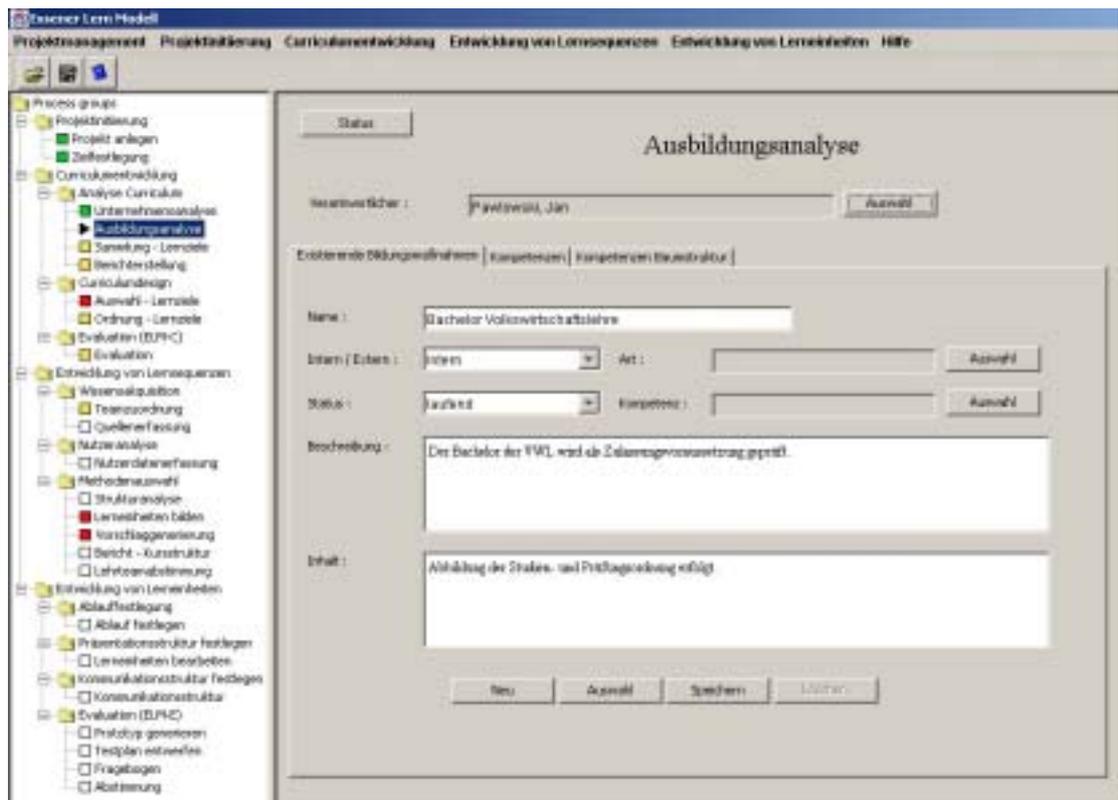


Abbildung 68: ELM-C Ausbildungsanalyse

Die Kontextanalyse wird fortwährend angepasst. Werden also spezifische Lernumgebungen konzipiert, so sollten die bereits bestehenden Ausbildungsmaßnahmen in diesem Bereich detailliert spezifiziert werden. In der Weiterentwicklung von ELM ist eine direkte Übernahme aus Personalentwicklungsdaten bzw. Ausbildungskatalogen geplant (vgl. Abschnitte 4.4.4 und 4.4.6.3).

Curriculumdesign

In der Prozessgruppe Curriculumdesign beginnt nun die Planung des Curriculums. Diese kann wiederum auf verschiedenen Aggregationsniveaus erfolgen. Einerseits kann an dieser Stelle ein gesamtes Curriculum für eine spezielle Ausbildungsmaßnahme entwickelt werden, andererseits kann es auch auf eine spezifische Kursentwicklung beschränkt sein.

Zunächst wird im Menü *Curriculumentwicklung*→*Curriculumdesign*→*Sammlung Lernziele* die Struktur der Lernziele festgelegt (strategische Ziele, Grob- und Feinziele). Diese können hierarchisch (*übergeordnete bzw. untergeordnete Lernziele*) oder durch die Spezifikation von *Voraussetzungen* verknüpft werden. Zudem werden die einzelnen Lernziele durch die in Abschnitt 5.4.3 beschriebenen Attribute charakterisiert (siehe Abbildung 69). Als Ergebnis dieser Phase entsteht ein Netzwerk von Lernzielen, das als Basis für eine Verhandlungsphase dient. Dieser Verhandlungsprozess wird im Menü *Curriculumentwicklung*→*Curriculumdesign*→*Auswahl Lernziele* weitergeführt. Zunächst werden die Lernziele priorisiert, dann in Verbindung zu Lernobjekten geordnet. An dieser Stelle kann bereits ein Bericht erstellt werden und eine erste Konsensabstimmung im Menü *Curriculumentwicklung*→*Evaluation* erfolgen.

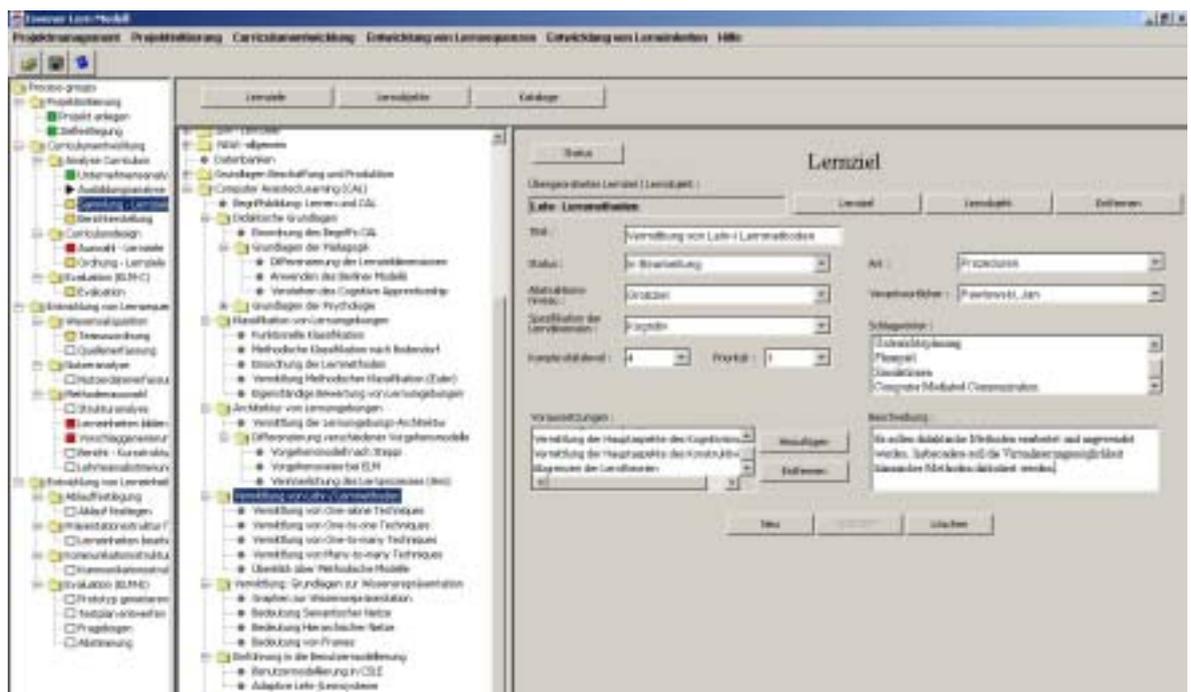


Abbildung 69: ELM-C Lernziele

Parallel zur Erfassung von Lernzielen wird eine grobe Strukturierung der Inhalte im Menü *Curriculumentwicklung*→*Analyse Curriculum*→*Sammlung Lernziele* vorgenommen (siehe Abbildung 70). Dabei werden zunächst inhaltlich gegliederte Lernobjekte erstellt, wobei die Aggregationsebene (Kurs, Zusammengesetzte Lerneinheit, Lerneinheit, Mediengruppe, Medienobjekt) an dieser Stelle nicht festgelegt werden muss, da diese erst bei einer genau-

eren Betrachtung der Inhalte durchgeführt werden kann. Zu jedem Lernobjekt können Metadaten nach LOM angelegt werden; auch diese Spezifikation wird jedoch in der Regel später im Entwicklungsprozess erfolgen. Um den Spezifikationsaufwand zu verringern, können Metadaten bereits an dieser Stelle an untergeordnete Lernobjekte vererbt werden (*Daten übernehmen*), um bereits Basisdaten (z. B. Autor, Beziehungen, Schlagworte) zu erfassen.

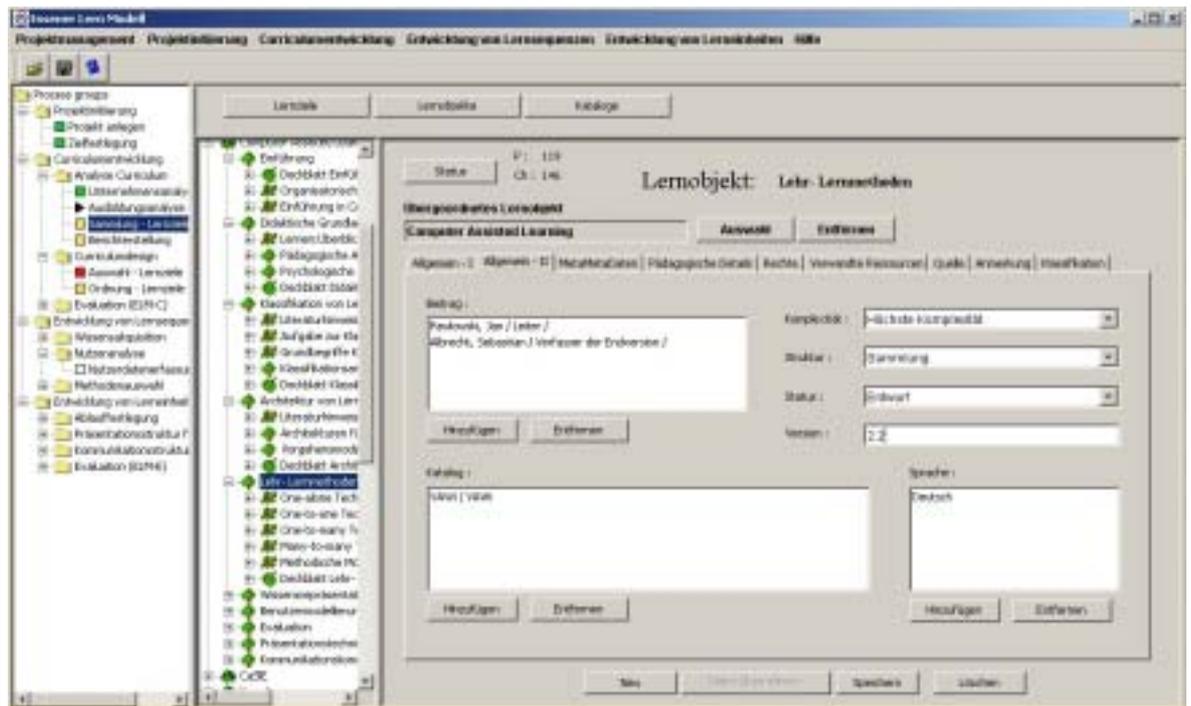


Abbildung 70: ELM-C Lernobjekte

Die Beziehungen zwischen den Lernobjekten und Lernzielen werden im Menü *Curriculumentwicklung* → *Curriculumdesign* → *Ordnung Lernziele* festgelegt. In dieser Phase werden Lernziele Lernobjekten zugeordnet. Die Darstellung in ELM erfolgt in einer Baumstruktur, jedoch können Lernziele auch mehreren Lernobjekten zugeordnet werden. Dieses ist gerade bei sozialen Lernzielen sinnvoll, die in verschiedenen Lernobjekten erfüllt werden sollen.

Evaluation

Zum Abschluss von ELM-C erfolgt im Menü *Curriculumentwicklung* → *Evaluation* die initiale Verhandlungsphase (siehe Abbildung 71). Aus den zuvor grob strukturierten Inhalten und Lernziele kann ein erster Prototyp erstellt werden, der die Navigationsstruktur und eine inhaltliche Gliederung enthält. Hinzu kommt die Evaluation der Inhalte und Lernziele, die mit der Konsensfeststellung endet. Entsteht kein Konsens, werden Änderungsanweisungen spezifiziert und der Prozess der Curriculumentwicklung wird erneut durchlaufen.

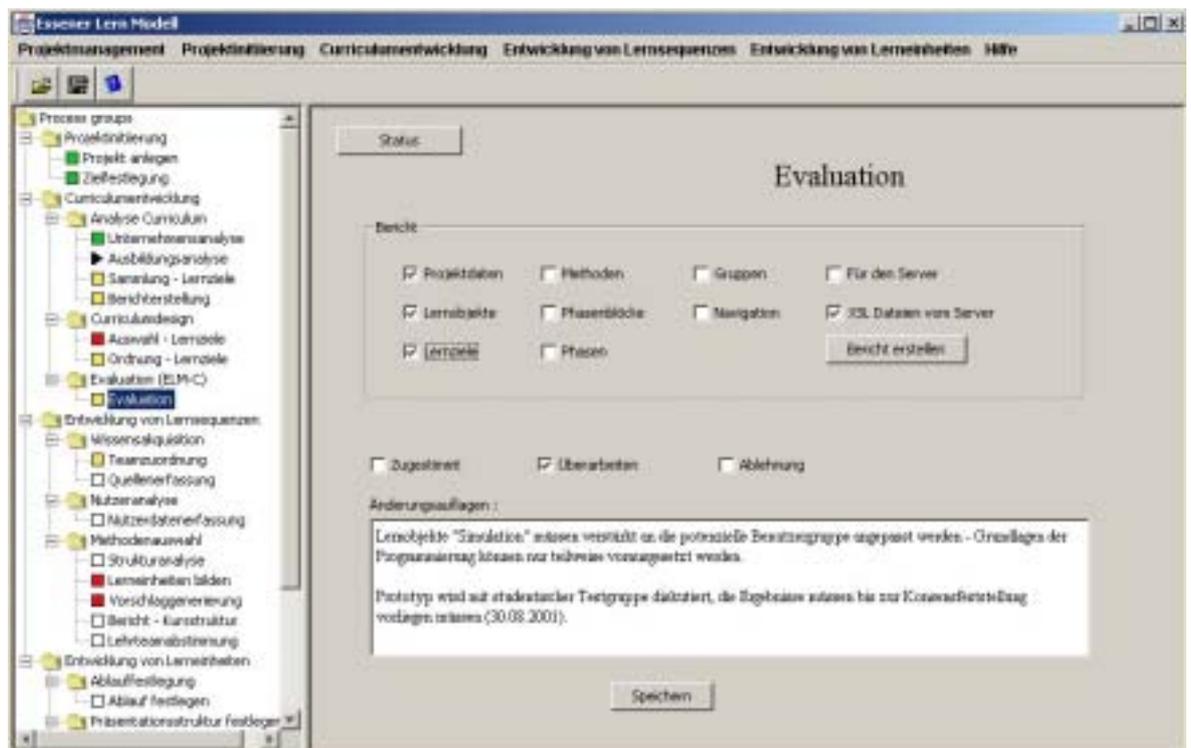


Abbildung 71: ELM-C Evaluation

6.1.2.3 ELM-D

In ELM-D erfolgt die Entwicklung von Lernsequenzen und somit das Design von Lernumgebungen. Die maßgeblichen Ergebnisse von ELM-D können wie folgt zusammengefasst werden (siehe 5.5):

- Quellenverzeichnis und -bewertung
- Analyse der potenziellen Lernenden, Lerngruppen und Lehrenden
- Modelle didaktischer Methoden
- Didaktische und inhaltliche Gestaltung der Lernumgebung
- Vorschlag zur Gestaltung der Präsentations-, Kommunikations- und Evaluationsstruktur
- Design der Lernumgebung
- Prototypische Lernumgebung
- Evaluation

Wissensakquisition

Die Phase der Wissensakquisition (*Entwicklung von Lernsequenzen* → *Wissensakquisition*) besteht zunächst aus einer Teamzuordnung zu den einzelnen Lernobjekten. Diese Teams werden sich in der Regel aus Entwicklern, Didaktikern und Inhaltsexperten zusammensetzen. Jedem Lernobjekt wird ein Team bzw. ein einzelner Akteur zugewiesen, der verantwortlich für die Bearbeitung eines Lernobjektes ist. Wird den jeweils untergeordneten Lernobjekten kein Verantwortlicher zugewiesen, so besteht die Verantwortlichkeit für alle untergeordneten Lernobjekte.

Darauf folgt die Quellenbearbeitung, die mit einer Recherchephase beginnt (Abbildung 72). Während dieser Recherche werden jeweils die Quellen, die Art der Quelle, die Auffindbarkeit (z. B. Bibliothekssignatur) und eine Bewertung vorgenommen. Die Quellenbewertung ist Teil einer Evaluation, so dass jeweils sichergestellt wird, dass die Quellen aktuell und inhaltlich adäquat sind.

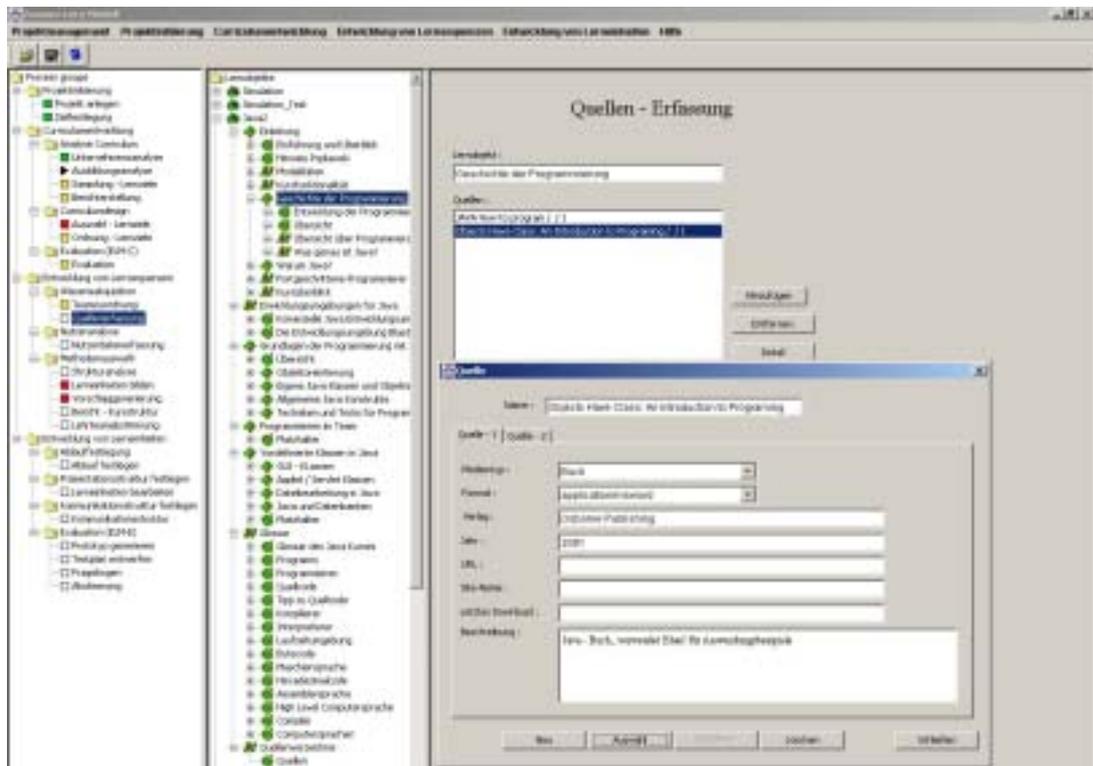


Abbildung 72: ELM-D Quellenbearbeitung

Benutzeranalyse

Parallel zum Prozess der Wissensakquisition (*Entwicklung von Lernsequenzen* → *Nutzeranalyse*) erfolgt die Analyse der Nutzer der Lernumgebung. Dabei wird die Analyse nicht auf Lernende beschränkt, vielmehr werden auch die Nutzerdaten von Lehrenden und Lerngruppen erfasst (Abbildung 73).

Für den Lehrenden werden neben persönlichen Daten insbesondere Erfahrungen und Kompetenzen erfasst, die bereits vorhanden sind. Diese beziehen sich auf Lernziele, Lernobjekte, Methoden und Softwareprodukte. Zusätzlich werden potenzielle Lerngruppen charakterisiert und bezüglich ihrer Zusammensetzung, räumlicher wie zeitlicher Verteilung und ihrer Kompetenzen analysiert. Diese Analyse und korrespondierende Anpassungen können auch zu einem späteren Zeitpunkt (insbesondere zur Nutzungszeit) durchgeführt werden.

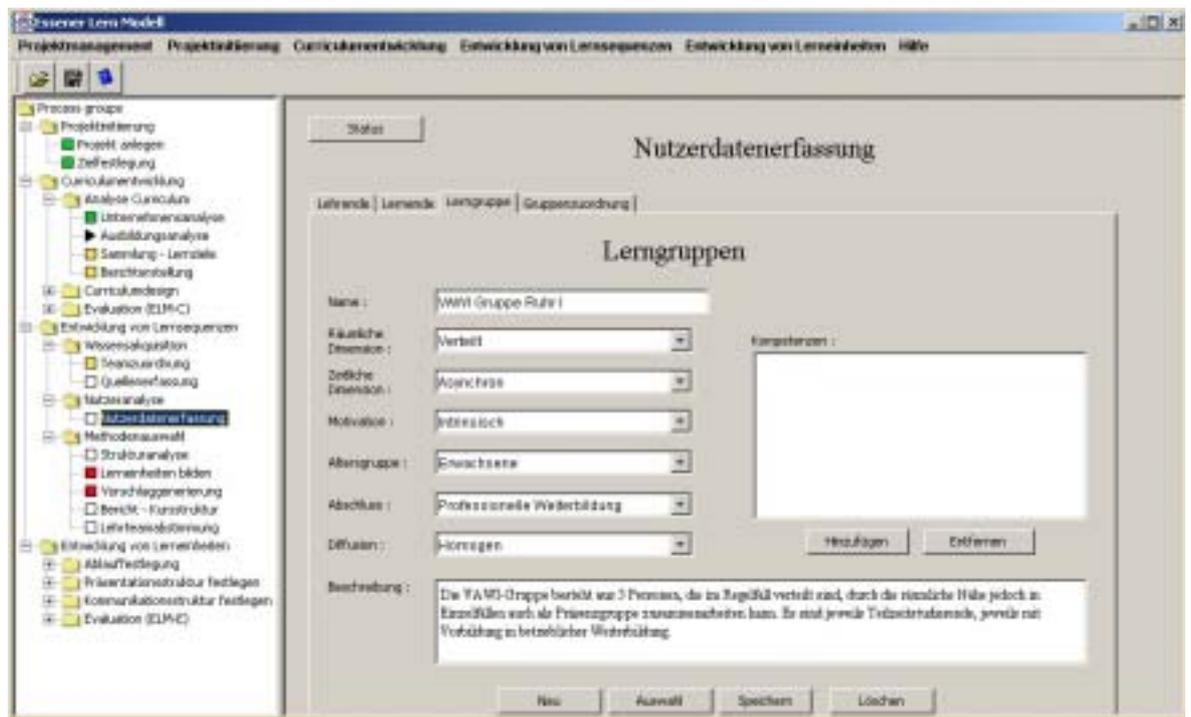


Abbildung 73: ELM-D Benutzeranalyse

Methodenauswahl

Die Prozessgruppe der Methodenauswahl bezieht sich auf die Modellierung didaktischer Methoden, die Bildung von Lerneinheiten und eine anschließende Methodenauswahl und -zuordnung.

Nach der umfangreichen Phase der Wissensakquisition werden die Lerneinheiten im Detail gegliedert und die entsprechende Lernzielzuordnung verfeinert. Dieser Prozess entspricht der Gliederung in ELM-C mit einem erhöhten Detaillierungsgrad. Es wird somit eine Einteilung in Kurse, Zusammengesetzte Lerneinheiten, Lerneinheiten, Mediengruppen und Medienobjekte vorgenommen.

Zentraler Bestandteil ist die Zuweisung bzw. Festlegung einer didaktischen Methode (*Entwicklung von Lernsequenzen*→*Methodenauswahl*→*Vorschlaggenerierung*). Voraussetzung für die Zuweisung ist die Modellierung einer didaktischen Methode. Dabei werden zunächst Eigenschaften einer Methode erfasst. Dies umfasst eine Beschreibung, Quelle der Methode, Erfahrungen mit einer Methode und eine Beschreibung, für welche Lernsituationen eine Methode angewendet werden kann bzw. Erfahrungen bestehen (Abbildung 74).

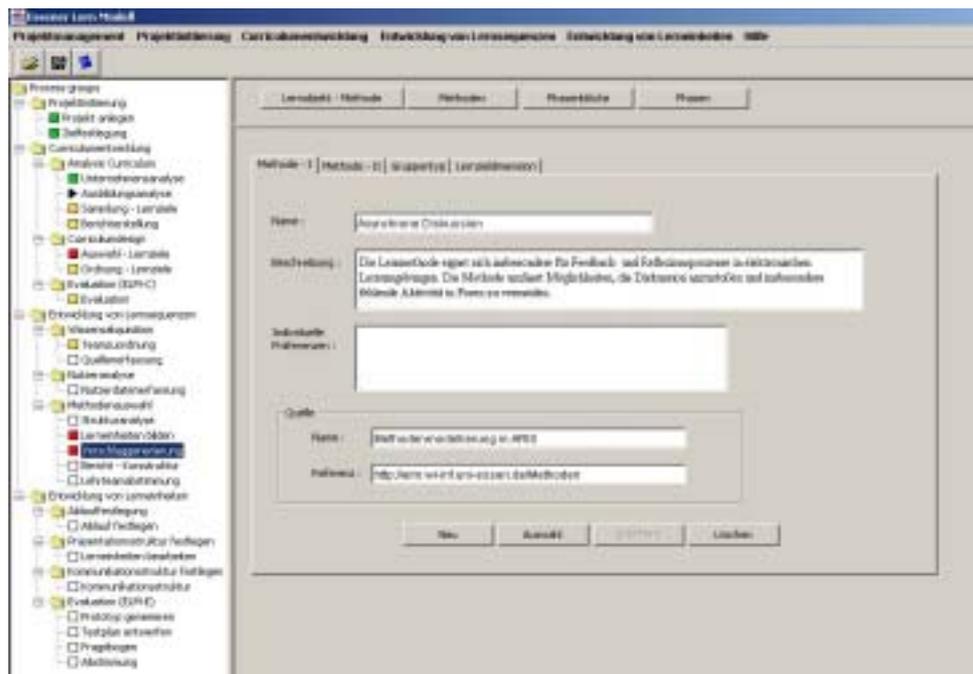


Abbildung 74: ELM-D Methodenmodellierung

Jede Methode kann aus verschiedenen Phasen bestehen (siehe Abbildung 75). Eine Phase wird neben einer allgemeinen Beschreibung durch verwendbare Informationsobjekte, die Beschreibung der Kommunikationsstruktur, Präsentationsstruktur und Akteurenrollen charakterisiert.

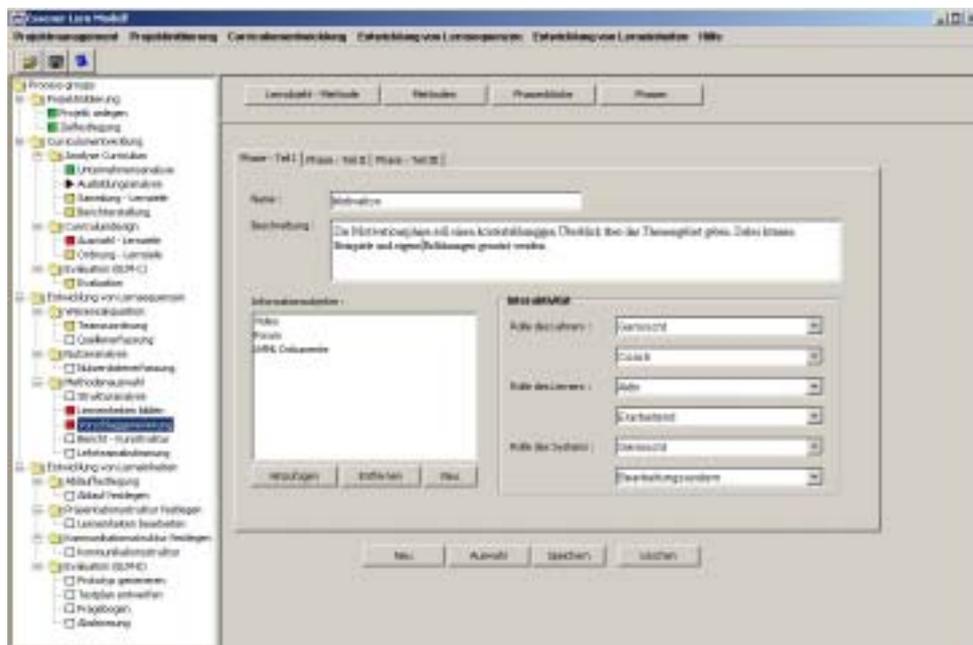


Abbildung 75: ELM-D Phasen einer Methode

Die einzelnen Phasen können zu Phasenblöcken zusammengefasst werden. Durch diese Art der Modellierung müssen Phasensequenzen, die wiederholt werden, nicht redundant modelliert werden (siehe Abbildung 76).

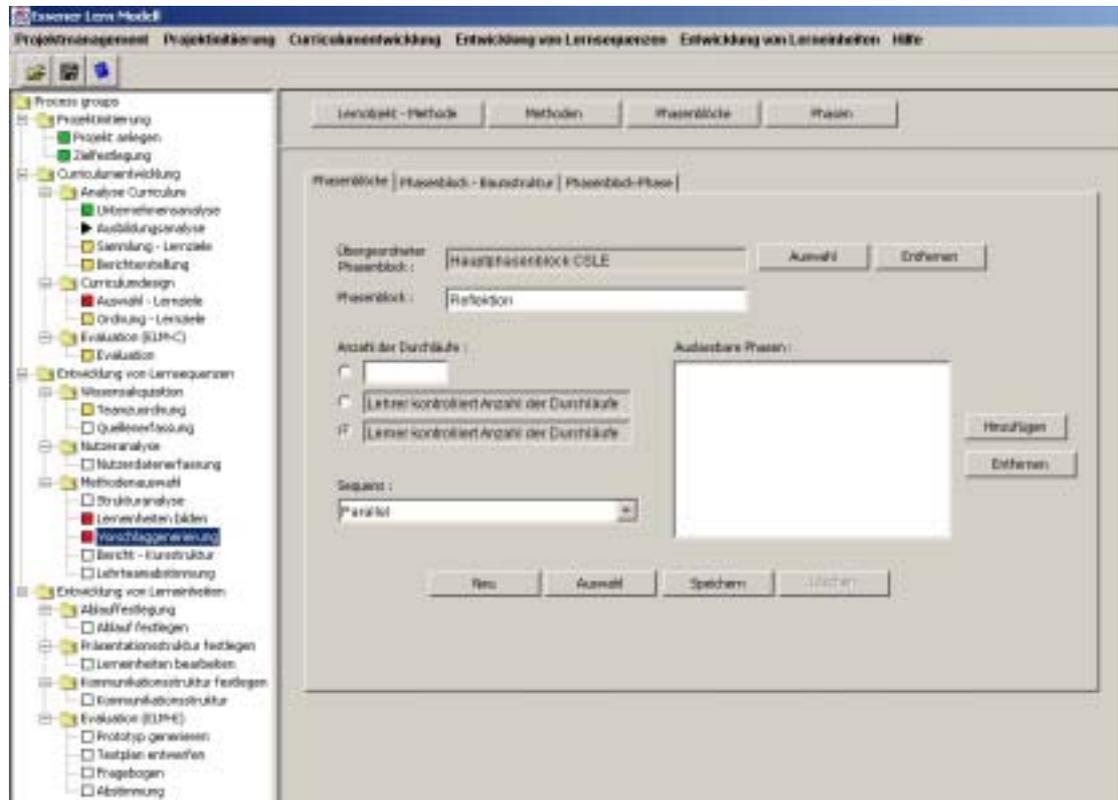


Abbildung 76: ELM-D Phasenblöcke einer Methode

Im Anschluss an die Modellierung erfolgt die Zuweisung einer Methode zu der Struktur der Lernobjekte. Eine Phase wird dabei jeweils einem Lernobjekt zugewiesen. Diese Zuweisung impliziert ein Design für die Lernumgebung, da mit der Methodenzuweisung gleichzeitig eine Interaktionsstruktur sowie Kommunikations-, Präsentations- und Evaluationskomponenten vorgeschlagen werden. Dieser Vorschlag kann durch den Entwickler genutzt oder aber modifiziert werden. Somit wird die Generierung von Lernumgebungen vereinfacht, ohne dabei jedoch die Kreativität einzuschränken.

An dieser Stelle werden zusätzlich die Metadaten nach LOM erweitert, da die Inhalte nun genauer beschrieben werden können. Ebenfalls können die LOM-Elemente zur Beschreibung der didaktischen Methode bearbeitet werden.

Nach diesen Prozessen wird ein umfangreicherer Prototyp generiert, der bereits die oben genannten Komponenten enthalten kann und somit einen Überblick über die Lernumgebung gibt. Die Evaluation wird durchgeführt und Änderungsanforderungen einbezogen bis schließlich die Konsensfeststellung erfolgt.

6.1.2.4 ELM-E

In ELM-E wird das Design der Lernumgebung im Detail bestimmt. Dabei werden auf der Ebene von Lerneinheiten Inhalte bearbeitet und eine genaue Beschreibung der Kommunikation und Evaluation gegeben (vgl. 5.6). Im Folgenden werden nur die in ELM-E zusätzlich spezifizierten Ergebnisse aufgeführt:

- Detaillierte Spezifikation der Lernumgebung
- Struktur der Lernumgebung durch die Komponenten Kurs, zusammengesetzte Lerneinheit, Lerneinheit, Mediengruppe, Medienobjekt
- Festlegung und Gestaltung der Präsentations-, Kommunikations- und Evaluationsstruktur
- Prototyp der Lernumgebung
- Bericht über verfügbare Metadaten
- Abschlussevaluation und –bericht

Ablauf festlegung

Die Ablauf festlegung bestimmt die Abfolge der Lerneinheiten und legt damit die detaillierte Struktur eines Kurses fest. Die graphische Oberfläche ermöglicht die Bearbeitung in Form einer Baumstruktur. Durch weitere Reihenfolgeoperatoren können zum Beispiel in explorativen Lernumgebungen verschiedene Lernwege frei von den Lernenden gewählt werden; Lernwege können aber auch vorgegeben werden.

Des Weiteren werden in dieser Phase a-priori-Anpassungen bezüglich der Lernerprofile spezifiziert. Derzeit kann auf die in CMI und in ELM-Lernerprofilen erfassten Daten zurückgegriffen werden. Die zusätzlichen Anpassungen müssen in den einzelnen Lerneinheiten durchgeführt werden. In 6.2.3 wird die Anpassung exemplarisch durch die Verwendung von Perl-Skripten beschrieben.

Lerneinheiten bearbeiten

Anschließend werden die Inhalte in das System eingepflegt. Dabei stehen generell zwei Alternativen zur Verfügung. Es können mit anderen Autorensystemen bearbeitete Inhalte einbezogen werden. Dabei steht eine Auswahlliste mit verfügbaren Formaten zur Verfügung, um diese Objekte in die Datenbank aufzunehmen. Ferner können Inhalte mithilfe des in ELM zur Verfügung stehenden Editor in die Datenbank übernommen werden. Diese Alternative stellt zusätzliche Beschreibungsmöglichkeiten durch die semantische Beschreibung mittels LMML zur Verfügung. Der Editor basiert auf dem Open-Source-Editor Jext [Guy2001]. Der ursprünglich für die Bearbeitung von Java-Programmen konzipierte Editor ermöglicht die Überprüfung von XML-Dokumenten bezüglich Wohlgeformtheit und Validität (Menüfunktion *Check Syntax*). Dafür wurde einerseits ein Plug-In implementiert, das diese Überprüfungen vornimmt, andererseits wurde eine DTD erstellt, die die möglichen Metadaten und Inhalte aus ELM enthält und die Validierung ermöglicht. Zusätzlich können Metadaten ebenfalls in diesem Editor bearbeitet werden. Dieses wurde als Zusatzfunktion implementiert, da gerade bei der Bearbeitung der Inhalte häufig Schlagworte oder Beschreibungen geändert werden und der Autor somit nicht zwischen Editor- und LOM-Ansicht wechseln muss. Abbildung 77 zeigt die Oberfläche zur Bearbeitung von Medienobjekten in ELM.

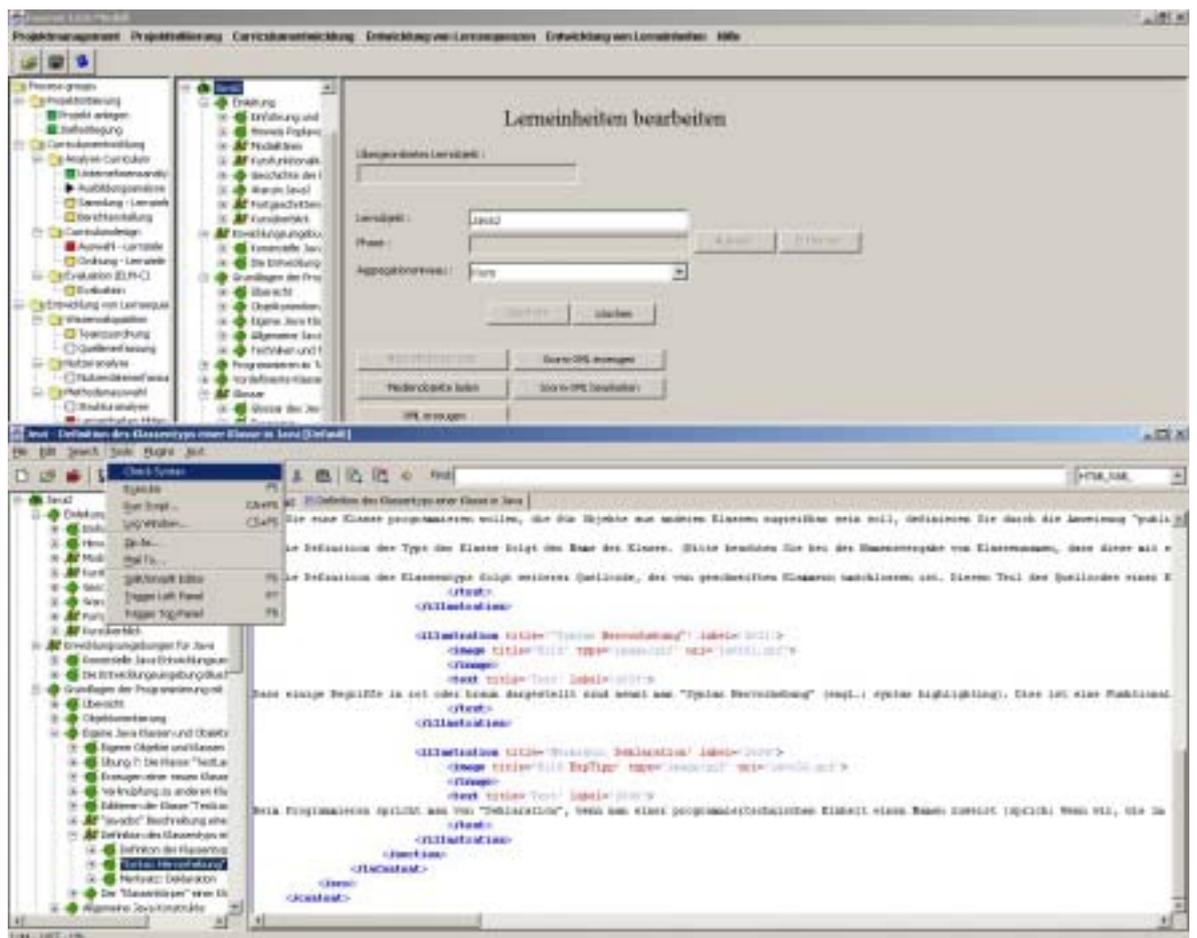


Abbildung 77: Editor zur Bearbeitung von Lerneinheiten

In ELM werden die Inhalte als Medienobjekte definiert und gespeichert. Textfelder werden direkt in der Datenbank gespeichert, während graphische Objekte als Files auf dem Server abgelegt und in der Datenbank referenziert werden. Des Weiteren ist eine Verknüpfung zu externen Dokumenten in Form verschiedener Links möglich (Abbildung 78).

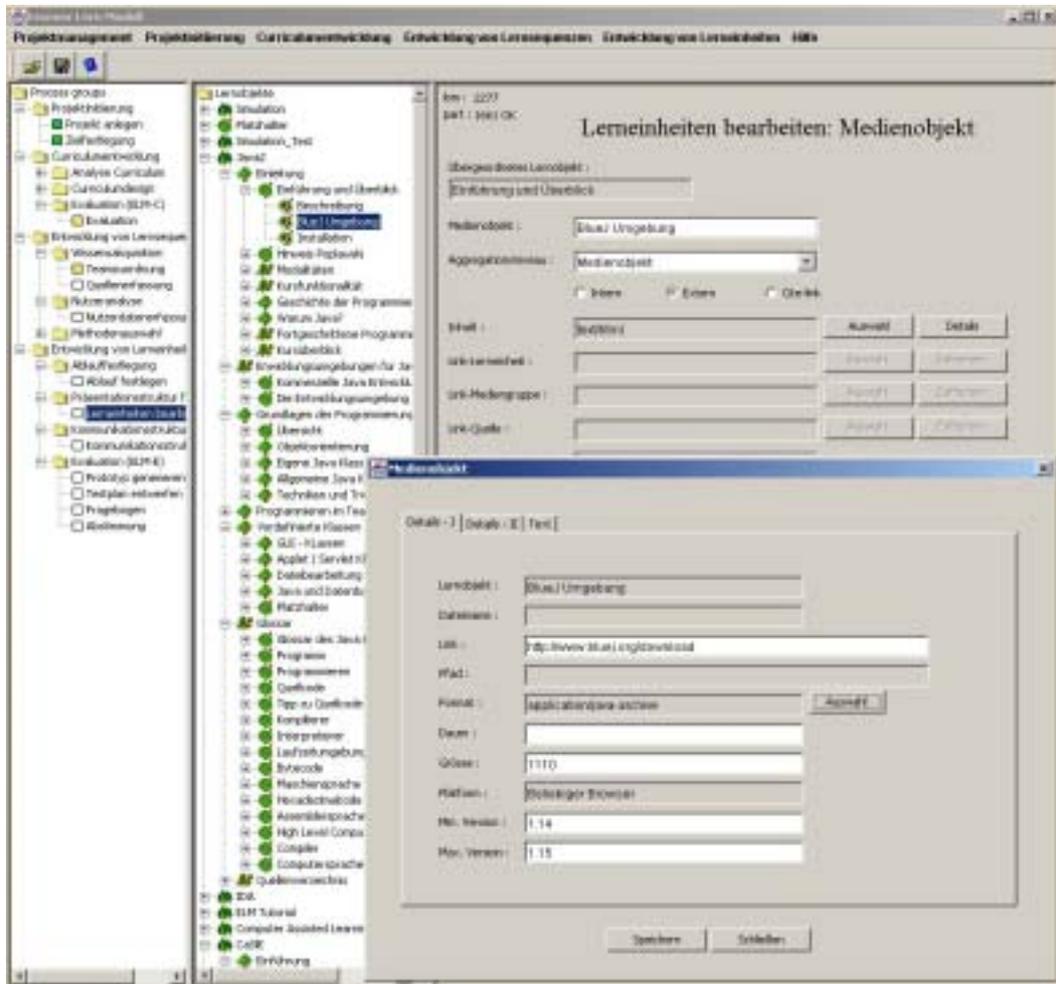


Abbildung 78: ELM-E Lerneinheiten bearbeiten, Medienobjekt

Mehrere Medienobjekte können zu einer Mediengruppe zusammengefasst werden, wenn sie noch nicht als eigenständig verwendbare Lernobjekte nutzbar sind. Die Spezifikation einer Mediengruppe ist in Abbildung 79 dargestellt.

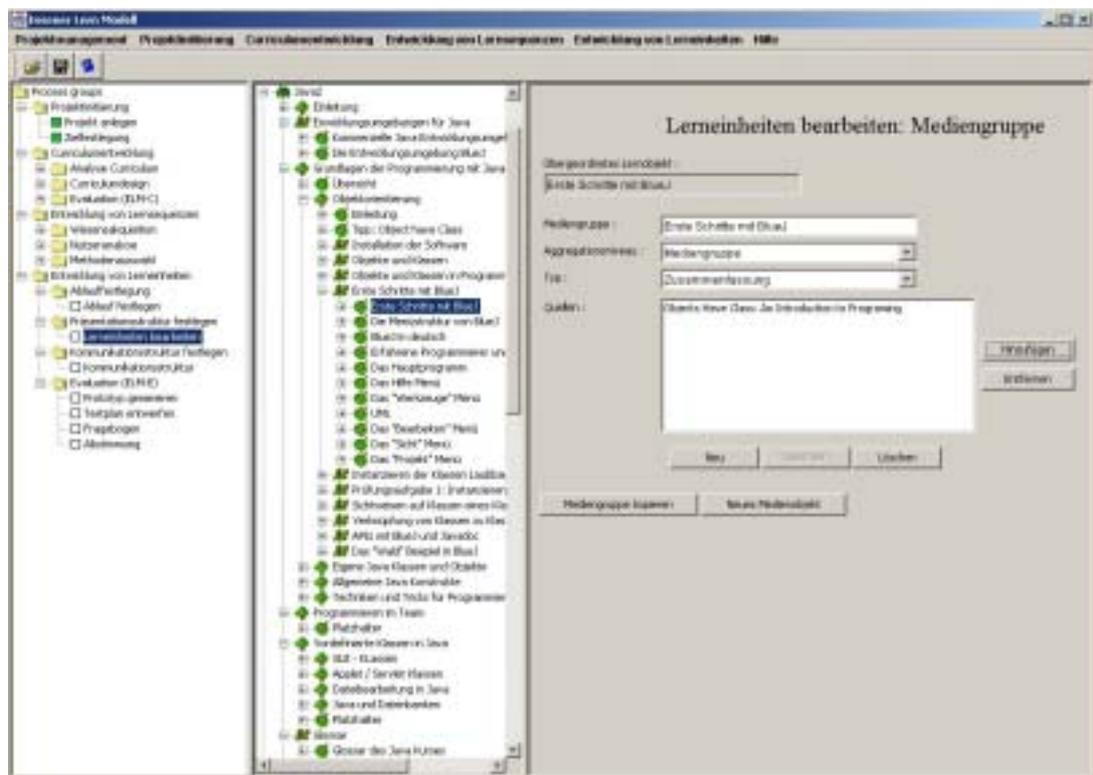


Abbildung 79: ELM-E Lerneinheiten bearbeiten, Mediengruppe

Parallel zu der Bearbeitung der Lerneinheiten werden Kommunikationsobjekte im Menü *Entwicklung von Lerneinheiten* → *Kommunikationsstruktur festlegen* spezifiziert (Abbildung 80). Ein Kommunikationsobjekt ist in der Regel eine Anwendung, die die Interaktion zwischen den Akteuren der Lernumgebung ermöglicht. Neben der Integration von Kommunikationsanwendungen wie Chat, Instant Messaging oder E-Mail können an dieser Stelle auch andere Interaktionsmöglichkeiten, wie zum Beispiel interaktive Übungen, einbezogen werden. Eine weitere Möglichkeit ist der Verweis auf Kommunikationsanwendungen in einem LMS, wenn die Steuerung der Lernumgebung über ein solches System vorgesehen ist.



Abbildung 80: ELM-E Kommunikation

Evaluation

In dieser Phase (*Entwicklung von Lerneinheiten*→*Evaluation*) werden zunächst verschiedene Möglichkeiten der Lernerfolgsüberprüfung zur Verfügung gestellt. Derzeit werden Multiple-Choice-Fragen, offene Fragen mit E-Mail-Einreichung und externe Testverfahren (z. B. Präsenzprüfung) unterstützt.

Nachdem in dieser Phase die Lernumgebung vollständig spezifiziert ist, wird eine Abschlussevaluation durchgeführt. Diese umfasst die Generierung eines Prototyps, die Generierung eines Berichts der verfügbaren Metadaten (siehe 6.1.3) sowie die Bereitstellung von Fragebögen bzw. Formularen zur Spezifikation von Änderungsanforderungen. Dies kann wiederum zu einer erneuten Bearbeitung der bisher beschriebenen Prozesse führen. Danach wird die abschließende Version der Lernumgebung generiert, die exemplarisch in den folgenden Abschnitten 6.2.1 bis 6.2.3 beschrieben wird.

Das Projekt endet zunächst mit der Konsensfeststellung für die Teil- und Endprodukte des Entwicklungsprozesses.

6.1.3 Metadaten-Navigator

Neben den Prototypen der Lernumgebung wird ein umfangreicher Bericht über die erfassten Metadaten in XML generiert. So kann einerseits ein Austausch zu anderen Repositories [vgl. HNWW2001] über eine direkte Datenbankanbindung erfolgen, andererseits können die verfügbaren Daten unter Verwendung der XML-Spezifikationen ausgetauscht werden.

Der Metadaten-Navigator enthält die wichtigsten Spezifikationen aus ELM, für die ein Navigationsframe generiert wird. Dabei sind folgende Daten enthalten:

- *Projekt*daten enthalten generelle Informationen über das Projekt wie Projektbeschreibung, Projektziele und die Kontextanalyse.
- *Struktur*daten beinhalten einerseits die ELM-Repräsentation der Lernobjekte, andererseits die SCORM-Repräsentation der Kurse.
- *Methoden*daten umfassen die Beschreibungen der vorhandenen Lernmethoden, der Phasenblöcke und Phasen der Methoden.
- *Gruppen*daten enthalten die Beschreibungen der Lerngruppen. Die individuellen Ler-nerdaten werden aus Datenschutzgründen nicht in den Bericht übernommen.
- Der Bericht kann für sämtliche Spezifikationen aus ELM generiert werden, so dass ein *individueller Bericht* erstellt wird.

Ein Beispiel für einen generierten Bericht ist in Abbildung 81 dargestellt.

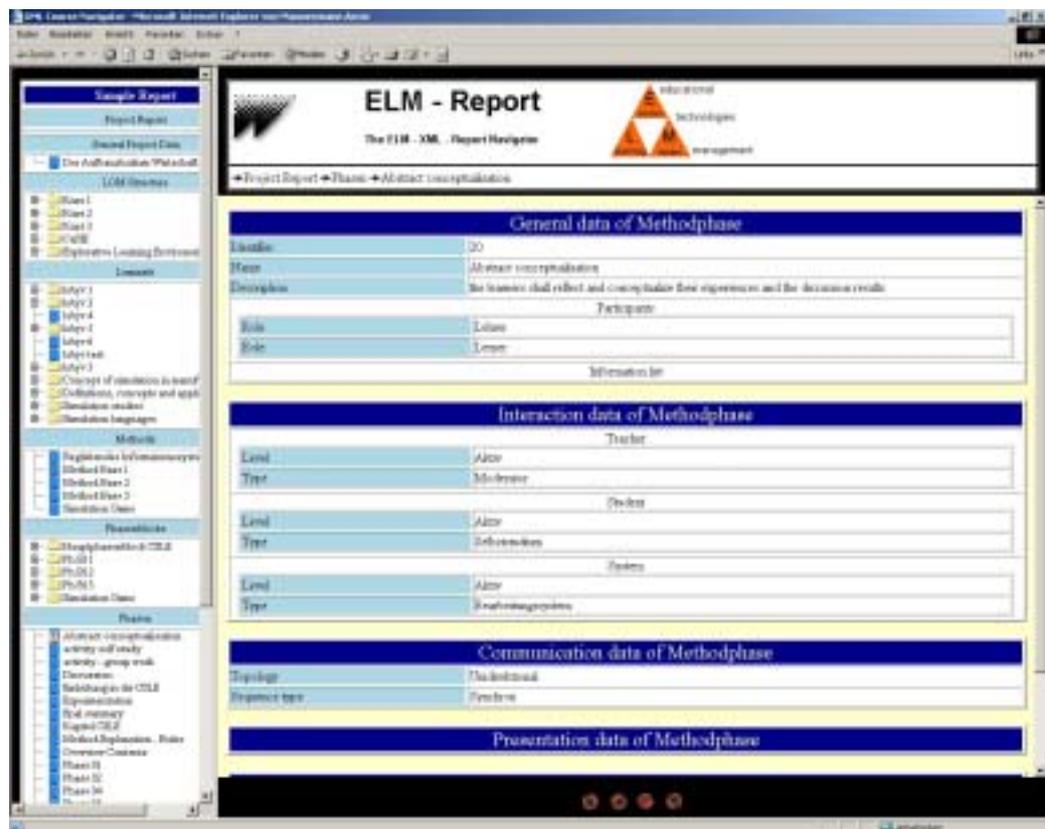


Abbildung 81: ELM-Bericht

6.2 Anwendungsbeispiele

Mithilfe des Essener-Lern-Modells sind bereits vielfältige Kursentwicklungen durchgeführt worden. Dabei wurden folgende Kurse für einen *Masterstudiengang Wirtschaftsinformatik* im Projekt *Virtuelle Aus- und Weiterbildung Wirtschaftsinformatik (VAWi)* [ABKP2001] und Lernumgebungen für das Fachgebiet *Wirtschaftsinformatik der Produktionsunternehmen* der Universität Essen implementiert:

- Einführung in die Simulation
- Integrierte Anwendungssysteme, Datenbanken, Archivierungssysteme
- Einführung in XML (siehe 6.2.1)
- Grundlagen der Programmierung, Programmierung in Java (siehe 6.2.2)
- XELE: XML-basierte Lernumgebung *Computer Assisted Learning* (siehe 6.2.3)
- MobiLum: Mobile Lernumgebung *Computer Assisted Learning* (siehe 6.2.3)

Die entwickelten Lernumgebungen werden sowohl im grundständigen Studium der Wirtschaftsinformatik an der Universität Essen als auch im virtuellen Weiterbildungsstudiengangs Wirtschaftsinformatik verwendet. Im Folgenden werden exemplarisch einige dieser Lernumgebungen vorgestellt. Eine detaillierte Darstellung der kompletten Entwicklungszyklen würde den Rahmen der Arbeit bei weitem überschreiten. Die Beschreibung erfolgt daher mit einem unterschiedlichen Detaillierungsgrad, um Redundanzen zu vermeiden und Besonderheiten der einzelnen Szenarien hervorzuheben:

- In Abschnitt 6.2.1 wird eine prototypische Lernumgebung mit dem Lerninhalt *XML* beschrieben. Die ELM-Applikation wurde hier zur Entwicklung einer studienbegleitenden Lernumgebung angewendet. Ziel des Einsatzes der Lernumgebung ist die Unterstützung einer Lehrveranstaltung des grundständigen Studiums.
- Im Abschnitt 6.2.2 wird die Verwendung von ELM in einem Szenario beschrieben, in dem sowohl Entwickler als auch Lernende verteilt arbeiten und somit ein wesentlich differierender Kontext gegeben ist. Es wird eine Lernumgebung mit dem Lerninhalt *Grundlagen der Programmierung* beschrieben, die sich wesentlich von derzeit verfügbaren Kursen aus diesem Bereich unterscheidet.
- Die Beschreibung von MobiLum in Abschnitt 6.2.3 beschreibt ein Szenario, in dem mobile Technologien zur Unterstützung von Lernprozessen eingesetzt werden. Der Schwerpunkt dieses Abschnitts liegt auf der Beschreibung der Methodenauswahl. Insbesondere der Einsatz neuer Technologien und der daraus resultierende Nutzen innerhalb des didaktischen Kontextes muss diskutiert werden.

6.2.1 Prototypische Entwicklung einer Lernumgebung XML

6.2.1.1 Anpassung

Die Anpassung der ELM-Applikation erfolgt spezifisch für die Entwicklung von Lerneinheiten im Fachgebiet *Wirtschaftsinformatik der Produktionsunternehmen (WiP)* der Universität Essen.

Zu Beginn des Entwicklungsprozesses werden zunächst alle der in Kapitel 5 beschriebenen Prozesse durchlaufen. Obwohl bereits ein festgelegtes Vorlesungsprogramm besteht, umfasst dieses auch die Prozesse der Curriculumentwicklung und Lernzielfestlegung. Der Grund für dieses Vorgehen ist die Tatsache, dass diese neue Spezifikation als Anlass genommen wurde, das bestehende Curriculum zu evaluieren, zu überarbeiten und formal zu spezifizieren. Diese Vorgehensweise bietet sich auch bei weiteren Projekten an, um die Aktualität des Curriculums und der Inhalte sicherzustellen. Die Terminologie des Essener-Lern-Modells wurde beibehalten, da keine universitätsinternen Richtlinien in dieser Beziehung vorgegeben sind.

Ein wichtiger Bestandteil der Curriculumentwicklung ist die Festlegung von Katalogen (zur Verwendung innerhalb der LOM). Die inhaltliche Einordnung erfolgt durch die Einordnung innerhalb des universitären Curriculums, das durch die Prüfungsordnungen der Studiengänge semiformal abgebildet ist. Dabei wurden die relevanten Studiengänge (Betriebswirtschaftslehre, Wirtschaftsinformatik, Volkswirtschaftslehre Bachelor) abgebildet und im *WiP-Katalog* erfasst. Im Rahmen des Weiterbildungsstudienganges VAWi werden die Kompetenzen der Studierenden, die als Voraussetzung einen Hochschulabschluss vorweisen müssen, im *VAWi-Katalog* abgebildet.

6.2.1.2 ELM-C

Projektinitiierung

Kern dieses Abschnitts ist die Erstellung einer XML-basierten Lernumgebung (XML-LU) mit dem Inhalt einer XML-Einführung. Die Aufgabenstellung impliziert den Entschluss, ein neues Teilcurriculum für eine solche Lehrveranstaltung zu entwickeln und zu implementieren, d. h. die *Projektinitiierung* ist erfolgt. Die Zielsetzung ist die Erweiterung und Flexibilisierung einer bestehenden Präsenzveranstaltung durch eine computerunterstützte Lernumgebung. Diese soll den Studierenden einerseits die Möglichkeit geben, Studieninhalte der Präsenzveranstaltung nachzuarbeiten, andererseits sollen interaktive Übungsmöglichkeiten geschaffen werden, die die Inhalte vertiefen und eigenständige Entwicklungsarbeit in XML fördern.

Kontextanalyse

Die *Kontextanalyse* erfordert zunächst eine Spezifikation der Strategien, Produkte und Services sowie verwendeter Standards innerhalb des Fachgebiets *Wirtschaftsinformatik der Produktionsunternehmen*. Die strategische Zielsetzung der universitären Ausbildung, die z. B. durch das Hochschulrahmengesetz oder die gesetzlichen Grundlagen des dualen Aus-

bildungssystem in Deutschland gegeben ist, wird dabei nicht näher spezifiziert. Vielmehr werden interne Zielsetzungen des Fachgebiets und des Fachbereichs erfasst. Dabei ist die Zielsetzung des Fachgebiets, Studierenden eine methodisch und theoretisch fundierte Ausbildung anzubieten, die die Basis sowohl für eine praktische Anwendung in Unternehmen als auch für wissenschaftliche Arbeit bietet. Die primären Produkte und Dienstleistungen des Fachgebiets sind dabei Lehrmaterialien und eine qualifizierte Beratung der Studierenden, wie auch die Forschung und Evaluation zur Qualität der Lehre. Des Weiteren werden im Fachgebiet verschiedene Vorlagen (Logos, Microsoft Word- bzw. LaTeX-Templates, Web-Präsentation) verwendet, die im Designprozess in die Layoutgestaltung eingehen.

Darauf folgt die Analyse der *bestehenden Bildungsmaßnahmen*, die jedoch bereits während der Anpassungsphase im WiP- bzw. VAWi-Katalog abgebildet wurden. Die Lernumgebung ist Teil der Lehrveranstaltung *Computer Assisted Learning II*. Somit bilden das Hochschulrahmengesetz sowie die Studien- und Diplomprüfungsordnungen den groben Rahmen dieser Veranstaltung. Als Bestandteil einer zwei Semesterwochenstunden umfassenden Lehrveranstaltung werden bei erfolgreicher Teilnahme im Rahmen der Diplomprüfungsordnungen auf Basis des Kreditpunktesystems zwei Kreditpunkte angerechnet. Die Lernerfolgsüberprüfung der gesamten Lehrveranstaltung erfolgt am Ende des Semesters in Form einer mündlichen Prüfung gemäß der gegenwärtig gültigen Fassung der Prüfungsordnung. Als grober Zeitrahmen für die Lernumgebung können ca. 12 Zeitstunden veranschlagt werden, wobei die Möglichkeiten von praktischen Übungen hier noch nicht berücksichtigt wurden. Zusätzlich erfolgt eine durch die Hochschule initiierte Evaluation durch eine Befragung der Studierenden zur Qualität der Lehrveranstaltung.

Im Anschluss an die Ausbildungsanalyse erfolgt die Analyse der *IT-Infrastruktur*. Die Anforderungen bez. der IT-Infrastruktur betreffen einerseits den Autor, andererseits die Lernenden. Die Anforderungen an die Informationstechnologie (IT) müssen als Mindestanforderungen (Tabelle 34) definiert werden, da nur so ein effektiver Arbeitsablauf gewährleistet werden kann. Die Mindestanforderungen werden für diesen Bereich sehr niedrig angesetzt, um das Ausgrenzen einzelner Studierender in verteilten Lernszenarien zu vermeiden. Insbesondere für Studierende der Wirtschaftswissenschaften können nur geringe IT-Kenntnisse vorausgesetzt werden, da im Gegensatz zu Studierenden der Wirtschaftsinformatik keine Grundlagen der Programmierung oder Datenbanken Bestandteil des Studiums sind. Es werden vor allem im Bereich der Software im Wesentlichen Produkte unterstützt, die frei verfügbar sind. Aufgrund der Tatsache, dass im Bereich XML noch sehr viel Entwicklungsarbeit erfolgt, kann der Anwender hier auf eine Vielzahl an Domain Specific Software (DSS) zugreifen (Editoren, Parser und Browser), die als Shareware oder als Evaluationsversionen erhältlich sind.

Tabelle 34: IT-Mindestanforderungen für Autoren und Anwender

| Autor: | | |
|------------------|---------------------------|---|
| Hardware | Prozessor: | Pentium III Klasse |
| | Arbeitsspeicher: | 64 MB |
| | Festplattenspeicher: | 10 GB |
| Netzwerk | Internetzugang: | ISDN (64 KBit/s) |
| Software | Betriebssystem: | Windows |
| | Präsentation: | MS Office 2000, Editor, Adobe Acrobat, Browser (IE 5.5, Netscape 4.7) |
| | Kommunikation: | E-Mail-Client |
| | Domain Specific Software: | XML-Editor, XML-fähiger Browser |
| Anwender: | | |
| Hardware | Prozessor: | Pentium II |
| | Arbeitsspeicher: | 64 MB |
| | Festplattenspeicher: | 5 GB |
| Netzwerk | Internetzugang: | Analog 56.6 KBit/s |
| Software | Betriebssystem: | Windows |
| | Präsentation: | Acrobat Reader 4.0, Editor, Browser |
| | Kommunikation: | E-Mail-Client |
| | Domain Specific Software: | XML-Editor, XML-Browser |

Als Hardwareausstattung stehen mehrere Server, Arbeitsplatzrechner sowie ein Rechnerpool für Studierende zur Verfügung, die jeweils den oben genannten Anforderungen entsprechen. Als Netzwerk des Fachgebiets bzw. der Universität steht ein 10 Based TX 100MBit (Fast Ethernet) zur Verfügung; die einzelnen Rechner dieses Netzwerkes können über das Universitätsnetzwerk sowie teilweise via Internet genutzt werden.

Das Einrichten der IT-Infrastruktur sowie die *Projektteambildung und Projektplanung* erfolgen parallel. Der Prozess der Teambildung kann verkürzt werden, da die Entwickler als Angestellte des Lehrstuhls bereits Erfahrungen in der Entwicklung von Lernumgebungen sowie der gemeinsamen Projektarbeit haben. Dies hat einerseits zur Folge, dass auch Fragen bez. der Kommunikations-, der Informations- und der Projektstruktur nicht näher betrachtet werden müssen. Andererseits muss das Risiko in Kauf genommen werden, dass der Arbeitsablauf aufgrund mangelnder Kontrollmöglichkeiten gestört wird.

Lernziele

Nach der Sammlung und Auswahl der Lernziele werden diese mithilfe des in Abschnitt 2.3.3 entwickelten Klassifikationsansatzes geordnet (siehe Tabelle 35). Im Folgenden wird die für die XML-LU erstellte Lernzielmatrix kurz erläutert:

- Die Dimension eines Richtzieles kann nur auf der sozialen Ebene klassifiziert werden, da die kognitiven und affektiven Bereiche durch die Dimensionen der einzelnen Grob- und Feinziele bestimmt werden.
- Consulting Skills werden insbesondere durch Motivationstechniken, Informations- und Präsentationstechniken, Führungs- und Organisationswissen sowie Rhetorikkenntnisse beschrieben.
- Die Dimension einiger Ziele ist sowohl kognitiv als auch affektiv, d. h., es soll hier einerseits prozedurales Wissen vermittelt werden, andererseits soll das Interesse des Lernenden am Themenbereich geweckt werden. Ferner sollen Lernende bei der Bildung von Werturteilen unterstützt werden.
- Die Klassifikationsebenen *Komplexität* und *Inhalt* sind hierarchisch aufgebaut; beispielsweise beinhaltet die Stufe Anwenden sowohl Verstehen als auch Kenntnis (vgl. 2.3.3).

Tabelle 35: Lernzielmatrix XML

| Richtziel | Abstraktionsgrad | | Dimension | Komplexität | Inhalt |
|---|--|------------------------------|-------------------|-------------|---------------------|
| | Grobziel | Feinziel | | | |
| Erkennen der Bedeutung und Möglichkeiten von Beschreibungssprachen für internet-unterstützte Lernumgebungen am Beispiel von XML | <i>Sozial:</i> Consulting Skills Teamfähigkeit | | | | komplexe Situation |
| | Markup-Sprachen | | kognitiv/affektiv | Verstehen | kontextfreie Fakten |
| | | Markup-Ausprägungen | kognitiv | Verstehen | kontextfreie Fakten |
| | | SGML | kognitiv | Verstehen | kontextfreie Fakten |
| | XML-Einführung | | kognitiv/affektiv | Verstehen | kontextfreie Fakten |
| | | XML-Grundlagen | kognitiv | Kenntnis | kontextfreie Fakten |
| | | SGML | kognitiv | Verstehen | kontextfreie Fakten |
| | | XML-Anwendungen | kognitiv/affektiv | Verstehen | kontextfreie Fakten |
| | XML-Syntax | | kognitiv | Synthese | Mustererkennung |
| | | Elemente (logische Struktur) | kognitiv | Anwenden | Problemlösen |

| Abstraktionsgrad | | | Dimension | Komplexität | Inhalt |
|------------------|----------------------------------|-----------------------------------|--------------------|-------------|---------------------|
| Richtziel | Grobziel | Feinziel | | | |
| | | Unicode (Zeichensatz) | kognitiv | Anwenden | Problemlösen |
| | | Entities (physikalische Struktur) | kognitiv | Anwenden | Problemlösen |
| | | Dokumententypen | kognitiv | Anwenden | Problemlösen |
| | | Wohlgeformtheit und Gültigkeit | kognitiv | Anwenden | Problemlösen |
| | Präsentieren von XML | | kognitiv | Synthese | komplexe Situation |
| | | Stylesheets | kognitiv | Anwenden | Problemlösen |
| | | XSL | kognitiv | Anwenden | Problemlösen |
| | Verarbeiten von XML | | kognitiv | Synthese | komplexe Situation |
| | | Parser | kognitiv | Anwenden | Problemlösen |
| | | DOM | kognitiv | Anwenden | Problemlösen |
| | XML-Erweiterungen | | kognitiv | Anwenden | Problemlösen |
| | | XML Namespaces | kognitiv | Verstehen | kontextfreie Fakten |
| | | XLink | kognitiv | Verstehen | kontextfreie Fakten |
| | | XPointer | kognitiv | Verstehen | kontextfreie Fakten |
| | | XQL | kognitiv | Verstehen | kontextfreie Fakten |
| | Werkzeuge für die Arbeit mit XML | | kognitiv/ affektiv | Bewerten | komplexe Situation |
| | | XMetaL | kognitiv | Analyse | Mustererkennung |
| | | InDelv XML Client | kognitiv | Analyse | Mustererkennung |
| | | IBM XSL-Editor | kognitiv | Analyse | Mustererkennung |

Abschließend bleibt festzuhalten, dass die Ordnung der Lernziele trotz der Unterstützung des Klassifikationsansatzes aufgrund der Struktur des Projektteams als subjektiv verstanden werden muss, da die Lernziele nur innerhalb des gewählten Kontextes klassifiziert wurden.

Innerhalb dieses Abschnitts wird die prototypische *Implementierung* des Teilcurriculums bzw. der XML-LU betrachtet. Die innerhalb des Entwicklungsprozesses gewonnenen Erkenntnisse müssen in die Modifikation der XML-LU fließen, so dass eine Umsetzung des Curriculums erfolgen kann.

Die *Evaluation* des Curriculums erfolgt im Test- und Realbetrieb. Eine Revision wurde nach einer Testphase durchgeführt. Die Lernerfolgskontrolle ist generell als mündliche Prüfung zum Ende der Vorlesungszeit geplant. Daneben ist es Ziel, im Rahmen der Lernumgebung bereits erbrachte Leistungen, wie Übungen, hier zu berücksichtigen.

6.2.1.3 ELM-D

Wissensakquisition

Im Rahmen der *Wissensakquisition* wurden relevante Quellen identifiziert, auf die die Lerninhalte der XML-LU aufbauen. Als Grundlage der Lernumgebung wurden insbesondere fünf Ressourcen identifiziert:

- XML Recommendation 1.0, Second Edition [WWWC2000]
- XML Handbuch [GoPr1999]
- Hypermedia: Konzepte und Sprachen im WWW (Skript TUM) [Teeg1999]
- XML Kompakt [Mich1999]
- XML Praxis und Referenz [PoWi2000]

Prinzipiell stellt die *XML Recommendation*, die Empfehlung des World Wide Web Consortiums (W3C), die Basis für alle Ressourcen in diesem Bereich dar. Neben den weiteren verwendeten (Lehr-)Büchern, *XML Handbuch*, *XML Kompakt* und *XML Praxis und Referenz*, handelt es sich bei der letzten Ressource (*Hypermedia: Konzepte und Sprachen im WWW*), um ein Skript der Technischen Universität München, welches dort bereits zu Ausbildungszwecken verwendet wird.

In Tabelle 36 wird die innerhalb der ELM-Klassifikation der Ressourcen am Beispiel der XML Recommendation 1.0, Second Edition [WWWC2000] dargestellt (vgl. auch Abschnitt 5.5.1).

Hinzu kommt die Expertise der Anwender. In Befragungen wurden Domänenexperten aus dem Fachbereich einbezogen und bezüglich des Themengebietes in den Entwicklungsprozess involviert. Dabei wurden insbesondere Anwendungserfahrungen abgebildet, die ebenfalls in den Kurs eingehen.

Tabelle 36: Spezifikation der XML Recommendation

| Attribut | XML Recommendation 1.0 |
|-----------------------|--|
| Allgemein | <p><i>Titel/Katalogeintrag:</i></p> <p>Extensible Markup Language (XML) 1.0, W3C Recommendation 2000-08-14</p> <p><i>Sprache:</i></p> <p>Englisch</p> <p><i>Beschreibung:</i></p> <p>Die Extensible Markup Language ist eine Teilmenge der Standard Generalized Markup Language (SGML), welche vollständig in diesem Dokument beschrieben wird. Das Ziel ist es, generisches SGML in der Weise über das Web auszuliefern, zu empfangen und zu verarbeiten, wie es jetzt mit HTML möglich ist. XML wurde entworfen, um eine einfache Implementierung und Zusammenarbeit sowohl mit SGML als auch mit HTML zu gewährleisten.</p> |
| Lebenszyklus | <p><i>Status:</i></p> <p>Empfehlung des W3C (stabiles Dokument) - Version 1.0</p> <p><i>Autoren:</i></p> <p>Tim Bray, Jean Paoli und C. M. Sperberg-McQueen</p> |
| Meta-Metadaten | LOM 6.0 |
| Technisch | <p>Formate:</p> <p>HTML, PDF, PS, XML</p> |
| Pädagogisch | Pädagogische Merkmale können für diese Ressource nicht festgelegt werden. Der Inhalt des Dokumentes ist auf einem hohen Niveau anzusiedeln. |
| Rechte | Das Dokument ist frei verfügbar. Die Rolle des W3C bei der Erstellung dieser Empfehlung ist es, die Spezifikation bekannt zu machen und ihre breite Anwendung zu fördern. |
| Beziehung | Die Ressource stellt die Basis für die anderen vier Ressourcen dar: XML Handbuch, XML Kompakt, XML Praxis und Referenz, Hypermedia: Konzepte und Sprachen im WWW |
| Erläuterung | Zurzeit keine Erläuterungen vorhanden |
| Klassifikation | Klassifikationsschema des W3C |

Im Anschluss an die Klassifikation der Ressourcen folgt die Zuordnung von inhaltlichen Schwerpunkten zu bestimmten Lernzielen. Dies kann im Falle der zu erstellenden Lernumgebung jedoch nur allgemein erfolgen, da die verschiedenen Konzepte im Zusammenhang mit verschiedenen Lernzielen stehen.

Tabelle 37 stellt das Ergebnis der Zuordnung exemplarisch dar.

Tabelle 37: Exemplarische Zuordnung von Ressourcen zu Lernzielen

| Abstraktionsgrad | | Ressource |
|------------------|-----------------------------------|--|
| Grobziel | Feinziel | |
| Markup-Sprachen | Markup-Ausprägungen | <ul style="list-style-type: none"> • Skript TUM • XML Praxis und Referenz |
| | SGML | <ul style="list-style-type: none"> • Skript TUM |
| XML-Einführung | XML-Grundlagen | <ul style="list-style-type: none"> • XML Praxis und Referenz • XML Kompakt • Skript TUM |
| | XML-Anwendungen | <ul style="list-style-type: none"> • XML Handbuch • XML Praxis und Referenz |
| XML-Syntax | Elemente (logische Struktur) | <ul style="list-style-type: none"> • XML Handbuch • XML Recommendation 1.0 |
| | Unicode (Zeichensatz) | <ul style="list-style-type: none"> • XML Handbuch • XML Recommendation 1.0 |
| | Entities (physikalische Struktur) | <ul style="list-style-type: none"> • XML Handbuch |
| | Dokumententypen | <ul style="list-style-type: none"> • XML Handbuch • Skript TUM • XML Kompakt |
| | Wohlgeformtheit und Gültigkeit | <ul style="list-style-type: none"> • XML Handbuch • Skript TUM • XML Kompakt |

Der Entwurf einzelner Lernsequenzen sowie deren Interdependenzen werden im Rahmen dieser Arbeit in den Phasen Methodenauswahl und Ablauffestlegung (ELM-D) betrachtet.

Benutzeranalyse

Die Nutzer der XML-LU sind Studierende der Wirtschaftsinformatik bzw. Wirtschaftswissenschaften an der Universität Essen. Die Lernumgebung ist Teil einer Hauptstudiumsveranstaltung; somit gilt als Voraussetzung für alle Nutzer der erfolgreiche Abschluss des jeweiligen Grundstudiums. Es bestehen jedoch wesentliche inhaltliche Unterschiede im Vorwissen der zwei Studierendengruppen. Ausgehend von der Struktur des Grundstudiums sind die Studierenden der Wirtschaftsinformatik bez. der Inhalte der XML-LU im Vorteil. Innerhalb des Studiums der Wirtschaftsinformatik liegt bereits hier ein Schwerpunkt auf Programmier- und Netzwerktechnologien; es wird eine breite Wissensbasis in der Domäne Informatik geschaffen. Dies impliziert, dass die Gruppen bei praxisorientierten, kooperativen Arbeiten Studierenden beider Studiengänge enthalten. Es sollten Gruppen gebildet

werden, die kein zu starkes Leistungsgefälle erwarten lassen. Aufgrund seiner wachsenden Bedeutung kann dennoch auch bei den Studierenden der Wirtschaftswissenschaften von fundierten Kenntnissen im Bereich Internet ausgegangen werden.

Bezüglich der Lehrenden kann auf umfangreiche inhaltliche, informationstechnologische und didaktische Erfahrungen zurückgegriffen werden. Insbesondere Methoden des Planspiels, der Simulation und des Coachings bei explorativen Lernmethoden gehören durchgängig theoretisch wie praktisch zu den Kompetenzen des Lehrpersonals. Hinzu kommen Kompetenzen bezüglich des virtuellen Coachings durch die Verwendung von Foren, E-Mail oder Instant Messaging-Anwendungen.

Methodenauswahl

Die räumliche und zeitliche Zuordnung der Nutzer kann, unterstützt durch die dem Fachgebiet und der Universität zur Verfügung stehende Hardwaresituation, vielschichtig erfolgen. Als Bestandteil einer herkömmlichen Veranstaltung über zwei Semesterwochenstunden können innerhalb der XML-LU synchrone Präsenzveranstaltungen, aber auch verteilte asynchrone Phasen unterstützt werden.

Die an Universitäten häufig verwendete Methode der Vorlesung wird oftmals den komplexen Lerninhalten im Bereich der IT nicht gerecht. Gute Erfahrungen wurden insbesondere mit der Kombination von verschiedenen Unterrichtsmethoden erzielt [ABPa00b]. Insbesondere durch Praxisbeispiele können auch andere Kompetenzen, wie Teamfähigkeit (Planspiel) oder Consulting Skills (Fallstudie/Planspiel), vermittelt werden. Prinzipiell werden Unterrichtsmethoden verwendet, mit denen bereits Erfahrungen gesammelt wurden:

- Vorlesung/Frontalunterricht
- Tutorium/Programmierübung
- Fallstudie/Planspiel
- Gruppenunterricht/Diskussion
- Exploratives Lernen/Individueller Lernplatz

Im Verlauf des ELM-Entwicklungsprozesses werden den im Rahmen dieser Arbeit klassifizierten Lernzielen Lernsequenzen zugeordnet; die erarbeiteten Grobziele werden in Lernsequenzen transformiert. Ergänzend werden den Klassifikationsmerkmalen der einzelnen Sequenzen sowie deren korrespondierenden Feinzielen anschließend Prioritäten zugewiesen. Dies erfolgt vor dem Hintergrund, den Auswahlprozess zu vereinfachen, da einzelne Lernziele mit komplementären Unterrichtsmethoden erreicht werden können.

So kann beispielsweise das Grobziel *XML-Einführung* bez. seiner Dimension nicht eindeutig definiert werden (vgl. 2.3.3.1.2). Die kognitive Ebene kann vor allem durch die Vermittlung prozeduralen Wissens erreicht werden. Veränderungen auf der affektiven Ebene können zwar durch das Lehren von Faktenwissen unterstützt werden, aber erst praktische Erfahrungen haben differenzierte Wertbetrachtungen zur Folge, was weiterhin auch die sozialen Lernziele fördert.

Für jede Sequenz werden die Merkmale mit der höchsten Priorität ausgewählt (5.4.3), wobei die Vergabe der Prioritäten für die einzelnen Merkmale nur kontextbezogen erfolgt.

Tabelle 38: Determinierung der Unterrichtsmethoden

| Sequenz | Priorisierte Merkmale | Unterrichtsmethode |
|----------------------------------|---|--|
| Markup-Sprachen | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Verstehen • <i>Inhalt:</i> kontextfreie Fakten | <ul style="list-style-type: none"> • Computerunterstützter Frontalunterricht/Vorlesung |
| XML-Einführung | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Verstehen • <i>Inhalt:</i> kontextfreie Fakten | <ul style="list-style-type: none"> • Computerunterstützter Frontalunterricht/Vorlesung |
| XML-Syntax | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Synthese • <i>Inhalt:</i> Mustererkennung | <ul style="list-style-type: none"> • Computerunterstützter Frontalunterricht/Vorlesung • Tutorium/Programmierübung |
| Präsentieren von XML | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Synthese • <i>Inhalt:</i> komplexe Situation | <ul style="list-style-type: none"> • Computerunterstützter Frontalunterricht/Vorlesung • Fallstudie |
| Verarbeiten von XML | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Synthese • <i>Inhalt:</i> komplexe Situation | <ul style="list-style-type: none"> • Computerunterstützter Frontalunterricht/Vorlesung • Planspiel |
| XML-Erweiterungen | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Verstehen • <i>Inhalt:</i> kontextfreie Regeln | <ul style="list-style-type: none"> • Gruppenunterricht/Diskussion |
| Werkzeuge für die Arbeit mit XML | <ul style="list-style-type: none"> • <i>Dimension:</i> kognitiv • <i>Komplexität:</i> Bewerten • <i>Inhalt:</i> Mustererkennung | <ul style="list-style-type: none"> • exploratives Lernen/ individueller Lernplatz |

Die gewählte Struktur der XML-LU kombiniert Unterrichtsmethoden, mit denen bereits Erfahrungen im Fachgebiet gemacht wurden. Diese Lösung bietet die Möglichkeit von synchronen Präsenzveranstaltungen und verteilten kollaborativen asynchronen Phasen. Auf eine genaue Spezifikation der jeweiligen Unterrichtsmethoden wird an dieser Stelle verzichtet und auf [Flech1996] und [Paul1995] verwiesen.

In Tabelle 38 wird deutlich, dass einige Lernsequenzen durch mehrere Unterrichtsmethoden unterstützt werden. Die Sequenz *Verarbeiten von XML* durchläuft erst eine Einführung in Frontalunterrichtsform, d. h. Vorstellen der theoretischen Konzepte, bevor in einem Planspiel das gelernte theoretische Wissen in einer praxisnahen Situation Anwendung finden soll. Ferner soll das Planspiel an dieser Stelle die Konzepte *XML-Syntax* und *Präsentieren von XML* vertiefen.

Design der Lernumgebung

Abschließend erfolgt das *Design der Lernumgebung*; hier wird aufbauend auf den Ergebnissen der bisher durchlaufenen Phasen die Grobstruktur der zu erstellenden Lernumgebung bez. des Designs entwickelt, insbesondere werden das Rahmenkonzept sowie sich daraus ergebende Abläufe festgelegt.

Den Studierenden soll unter Verwendung einer internetunterstützten Lernumgebung (IULU) ein flexibles Lernsystem zur Verfügung gestellt werden, ohne den administrativen Aufwand des Dozenten wesentlich zu steigern. Die XML-LU bietet den Studierenden die Möglichkeit, Fehlzeiten (verursacht z. B. durch Krankheit oder Nebentätigkeiten,) oder aber missverstandene Lerninhalte kompensieren bzw. nacharbeiten zu können. Daneben wird den Studierenden die komplexe Domäne XML durch die Verwendung einer IULU leichter zugänglich sein. Letztendlich unterstützt die IULU auch soziale Aspekte wie selbstverantwortliches Lernen oder in entsprechenden synchronen Phasen die Organisation von Lerngruppen.

Sowohl bei der isolierten Verwendung der IULU als auch bei der Verwendung in einem LMS erfolgt der Zugriff durch einen Browser. Durch die Nutzung des Internets als Basis der Lernumgebung, werden weitere Aspekte, wie z. B. Kommunikations- und Präsentationsmöglichkeiten, implizit festgelegt. Ferner kann mithilfe der IULU die gesamte Administration erfolgen (Kursanleitungen etc.).

Der innerhalb der IULU entstehende Aufwand für den Dozenten wird durch die Verwendung von XML-Dokumenten zur Repräsentation der Lerninhalte gering gehalten. XML unterstützt insbesondere die Trennung von Inhalt, Präsentation und Struktur eines Dokumentes (vgl. Abschnitt 4.3). Daraus folgend entsteht ein wesentlicher Vorteil gegenüber traditionell erstellten Lernmaterialien, wie Bildschirmpräsentationen oder Skripten. Die Lernmaterialien lassen sich bez. ihrer Struktur und neuer Inhalte wesentlich einfacher modifizieren, was gerade in der Domäne XML von entscheidender Bedeutung ist; ferner können Inhalte adaptiert und in anderen Lernszenarios wiederverwendet werden.

Auf Seiten der Lerner bietet die Unterstützung der unterschiedlichsten Repräsentationsformen die Möglichkeit, die Materialien den Präsentationspräferenzen sowie dem Lerntempo anzupassen.



Abbildung 82: Oberfläche XML-LU

Abbildung 82 zeigt die prototypische Oberfläche der XML-LU. Die in diesem Beispiel verwendeten Konzepte bez. der Präsentation (XSL) und der Struktur (DTD) wurden exemplarisch in Abschnitt 4.3 dargestellt.

Die resultierende Feinstruktur der IULU wird anschließend in ELM-E am Beispiel der Lerneinheit *Präsentieren von XML*-Fallstudie betrachtet.

6.2.1.4 ELM-E

Ablauffestlegung

Innerhalb der Designphase (ELM-D) wurde das Rahmenkonzept der XML-LU festgelegt. Darauf aufbauend erfolgt auf der Entwicklungsebene ELM-E die Realisierung der Feinstruktur der Lernumgebung. Die einzelnen Lernsequenzen werden in Lerneinheiten/Lernmodule dekomponiert (Abbildung 83), wobei anschließend der Ablauf sowie die Präsentations- und Kommunikationsstruktur der einzelnen Lerneinheiten festgelegt werden.

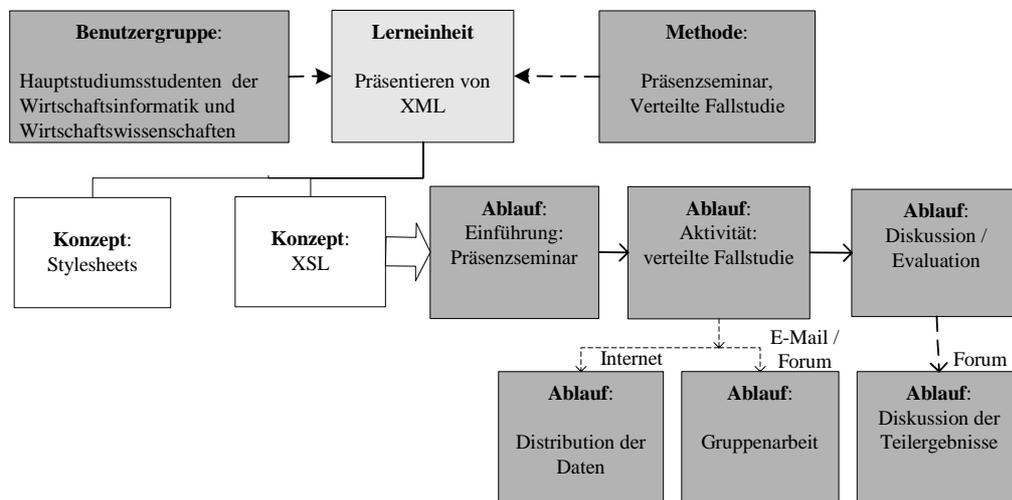


Abbildung 83: ELM-E: Präsentieren von XML

Als genereller Ablauf für die XML-LU lässt sich folgende Struktur festlegen. Dabei umfasst die Methode zu jedem Zeitpunkt ein begleitendes Informationssystem, das zur Unterstützung der Präsenzseminare verwendet wird.

Tabelle 39: Ablauf des XML-Kurses

| Inhalt | Methode |
|--------------------------------------|-----------------------|
| 1. Markup-Sprachen | (Präsenzseminar) |
| 1.1. Markup-Ausprägungen | |
| 1.2. SGML | |
| 2. XML-Einführung | (Präsenzseminar) |
| 2.1. XML-Grundlagen | |
| 2.2. SGML | |
| 2.3. XML-Anwendungen | |
| 3. XML-Syntax | (Präsenzseminar) |
| 3.1. Elemente | |
| 3.2. Unicode | |
| 3.3. Entities | |
| 3.4. Dokumententypen | |
| 3.5. Wohlgeformtheit und Gültigkeit | |
| 4. XML-Syntax | (Tutorium) |
| 5. Präsentieren von XML | (Präsenzseminar) |
| 5.1. Stylesheets | |
| 5.2. XSL | |
| 6. Präsentieren von XML | (Fallstudie) |
| 7. Verarbeiten von XML | (Präsenzseminar) |
| 7.1. Parser | |
| 7.2. DOM | |
| 8. Verarbeiten von XML | (Planspiel) |
| 9. XML-Erweiterungen | (Gruppenunterricht) |
| 9.1. XML Namespaces | |
| 9.2. XLink | |
| 9.3. Xpointer | |
| 9.4. Xquery | |
| 10. Werkzeuge für die Arbeit mit XML | (Exploratives Lernen) |
| 10.1. XmetaL | |
| 10.2. InDelv XML Client | |
| 10.3. IBM XSL-Editor | |

Im Laufe des ELM-Entwicklungsprozesses wurden der Lernsequenz *Präsentieren von XML* zwei Unterrichtsmethoden zugewiesen (vgl. 5.5.3). Einerseits erfolgt ein Präsenzseminar, um die Studierenden mit den entsprechenden Konzepten vertraut zu machen, andererseits wird eine Fallstudie verwendet, um die erlangten theoretischen Grundlagen durch praktische Anwendung zu vertiefen und zu verdeutlichen. Die Bearbeitung der Fallstudie ist als Gruppenarbeit geplant. Daraus folgend werden die Teilnehmer der Fallstudie auch in sozialer Hinsicht (Teamfähigkeit) geschult.

Lerninhalte bearbeiten

Im Folgenden werden die Ergebnisse aus ELM-E anhand der Fallstudie *Präsentieren von XML* (siehe Tabelle 39) exemplarisch beschrieben.

Im Rahmen einer Fallstudie bearbeiten die Lernenden einzeln oder in Gruppen rekonstruierte Praxisfälle, um sich Wissen über die betreffende Domäne anzueignen und ihre Urteils- und Entscheidungsfähigkeit auszubilden. Sie versetzen sich dabei in die Rolle von real handelnden Personen und müssen komplexe Zusammenhänge überblicken. Das verwendete Fallmaterial sollte die Situation möglichst genau und überschaubar dokumentieren, wobei die behandelten Beispiele häufig offene Entscheidungsalternativen besitzen. Fallstudien vermitteln somit auch Handlungs- und Entscheidungswissen [Flech1996]. Die Phasen der Fallstudie (Vorbereitung, Rezeption, Interaktion, Anwendung, Bewertung, vgl. 2.3.8.1) wurden zu drei Phasenblöcken zusammengefasst: Einführung, Aktivität/Interaktion, Analyse/Evaluation. Die Phasen wurden somit in ELM modelliert; die Terminologie wurde aufgrund bereits bestehender Methoden angepasst, da Analyse- bzw. Reflexionsphase wiederverwendet werden konnten.

Generell erfolgt in *der ersten Phase* die *Einführung* der Lernenden in den Themenbereich und, falls nötig, in die Fallstudienmethode [Flech1996]. Eine explizite Einführung ist innerhalb der XML-LU nicht notwendig, da dies bereits in den vorherigen Lernsequenzen erfolgte. Somit liegt der Fokus dieser Phase auf der Darstellung und Erklärung der Unterrichtsmethode Fallstudie und den sich daraus ergebenden Konsequenzen (selbstgesteuertes bzw. selbstverantwortliches Lernen). Im Anschluss an die Einteilung der Gruppen wird die Rahmenhandlung zur Fallstudie vorgestellt: Die Gruppen stellen eigenständige Unternehmen dar, die den Auftrag erhalten, einen Wissensinhalt für eine IULU in XML zu implementieren, wobei unterschiedliche Repräsentationsarten (Stylesheets), die in verschiedenen Unterrichtsmethoden verwendbar sind, erstellt werden.

Die Einführung wird als Präsenzveranstaltung geplant. Hier erfolgt die Erklärung der Regeln anhand einer Bildschirmpräsentation. Die Kommunikation erfolgt auf Diskussions- bzw. Nachfragebasis. Der Zeitrahmen ist auf eine Zeitstunde begrenzt, da die für die gesamte Fallstudie eingeplante Zeit insbesondere für Anwendungslernziele (Praxisanteil) genutzt werden soll.

Die Präsentation der Aufgabenstellung erfolgt in Form einer Videosequenz; dieses Konzept basiert auf der Methode der Anchored Instruction, einer Ausprägung der konstruktivistischen Ansätze des Lernens (vgl. Abschnitt 2.3.4.3). Ziel der Anchored Instruction ist es, mithilfe neuer Technologien wie z. B. Videopräsentationen, Lerninhalte in sinnvollen, problemorientierten und lebensnahen Kontexten zu verankern, so dass die Lernenden motiviert werden, selbstständig Probleme zu konstruieren, Themen zu explorieren und Lernergebnisse darzustellen [Hoff1997]. Die Kommunikation ist hier ebenfalls als direktes Gespräch geplant.

Abschließend erhalten die Studierenden einen Verweis (URI) auf die benötigten Ressourcen zur Bearbeitung der Fallstudie. Die Inhalte der Ressourcen werden mithilfe von XML-Dokumenten strukturiert; durch die Verwendung verschiedener Stylesheets können entweder WWW-Seiten erstellt oder als Ausdruck zur Verfügung gestellt werden. Der administrative Aufwand ist bei der ersten Durchführung der Fallstudie geringfügig erhöht, da für die verschiedenen Präsentationsformen Stylesheets entwickelt werden müssen. Dieser

Mehraufwand amortisiert sich jedoch bei einer erneuten Durchführung der Fallstudie. Ferner können die erstellten Stylesheets auch in anderen Lerneinheiten, z. B. dem Planspiel, verwendet werden.

Als Erweiterungsmöglichkeit ist eine Gruppe denkbar, die sich selbst organisiert und die gesamte Fallstudie verteilt bearbeitet (siehe auch 6.2.3). Durch die vorherige Bereitstellung einer CD mit allen wichtigen Materialien wird der noch häufig schlechten Netzanbindung der Teilnehmer Rechnung getragen. Alternativ wird eine Downloadmöglichkeit des Videos ermöglicht. Ferner könnte auch die Präsentation des Moderators zu Beginn der Einführungsveranstaltung ein Video enthalten. Die verteilt arbeitende Gruppe hat dann wie die anderen Gruppen die Möglichkeit, auf einen URI zuzugreifen, unter dem alle Unterlagen als XML-Dokumente bereitstehen. Die Kommunikation erfolgt via E-Mail für einzelne persönliche Fragen sowie zu festgelegten Terminen als Chat oder als Diskussionsforum. Letztere Alternative bietet mithilfe mehrerer Threads die Möglichkeit, den Ablauf der Kommunikation nachvollziehbar zu gestalten.

In der *Aktivitäts-/Interaktionsphase* bearbeiten und interpretieren die Studierenden das Fallmaterial. Falls nötig, werden zusätzliche Informationen selbstständig beschafft [Flech1996]. Der Lösungsweg ist nicht eindeutig vorgegeben, vielmehr erhalten die Studierenden die Möglichkeit zur Entfaltung ihrer Kreativität. Innerhalb der Gruppen kommt es zum Vergleich von Lösungsmöglichkeiten und Entscheidungsfindungen, um die Erfahrungen mit aktuellem Wissen in Beziehung zu setzen und daraus Schlussfolgerungen zu ziehen.

Es erfolgt keine weitere Präsentation von Inhalten seitens des Moderators; die Gruppen können auf alle bisher zur Verfügung gestellten Unterlagen zugreifen (URI) und zusätzliche Unterlagen beschaffen. Die Recherche weiterer Unterlagen wird durch eine separat zur Verfügung gestellte Linksammlung unterstützt; erstellt wird diese mithilfe der Extensible Linking Language (XLink) (vgl. 4.3). Die Umsetzung einer solchen Ressource ist zwar zunächst aufwendiger als in HTML, dafür wird jedoch die Wiederverwendbarkeit und Anpassung in anderen Zusammenhängen unterstützt.

Die XML-LU bietet den Studierenden die Möglichkeit, sich selbst zu organisieren: So können sie lokal (Rechnerpool des Instituts) oder verteilt arbeiten. Der Informationsaustausch wird seitens der IULU durch einen separaten Up- und Downloadbereich unterstützt, den die einzelnen Gruppen selbst verantwortlich pflegen. Ferner steht ein moderiertes Forum zur Verfügung.

Zu Beginn dieser Phase werden vom Moderator klar definierte Kommunikationstermine festgelegt, um so mögliche Fragen stellen und Missverständnisse klären zu können. Die Kommunikation erfolgt mithilfe eines Forums. Dadurch werden Fragen strukturiert und sind auch später für alle Teilnehmer nachvollziehbar. So kann sich in einer Gruppe eine Frage zu einem bestimmten Thema erst wesentlich später ergeben. Der Moderator wird durch das Forum unterstützt, da die Studierenden nach ähnlichen Problemen und deren Lösungen suchen können. Die verteilt arbeitenden Gruppen müssen sich mithilfe von E-Mail, Chat oder Videoconferencing verständigen und organisieren. Die Forumsbeiträge werden später zusammengefasst und für weitere Kurse in Form von FAQ bereitgestellt.

In der Phase *Analyse und Evaluation* werden die Lösungen der einzelnen Gruppen vorgestellt und diskutiert sowie ein Bezug zur Realität geschaffen. Die Äußerung konstruktiver Kritik an der Fallstudie durch die Teilnehmer wird gefördert, so dass die Fallstudie mithilfe der Lernererfahrungen verbessert werden kann.

Die einzelnen Gruppen stellen ihre Ergebnisse im Rahmen einer Präsenzveranstaltung vor; ein genereller Zeitrahmen ist hinsichtlich der gesamten Phase nicht zu bestimmen, da dies von der Anzahl der Teilnehmer und Gruppen abhängig ist. Es empfiehlt sich, diese Phase der Fallstudie als Präsenzveranstaltung durchzuführen, da der hohe administrative und technische Aufwand eine verteilte Veranstaltung erschwert.

Die Studierenden präsentieren ihre Ergebnisse in einem Vortrag von maximal zwanzig Minuten. Anschließend erfolgt eine Diskussion des Beitrages. Die Studierenden stellen ihre Ergebnisse mithilfe von XML-Dokumenten dar, um die Inhalte weiter zu vertiefen. Die Kommunikation innerhalb der Präsenzveranstaltung erfolgt durch Diskussionen, wobei der Moderator eine vorwiegend koordinierende Rolle einnimmt. Abschließend werden die Erfahrungen der Teilnehmer und des Dozenten diskutiert; es wird ein Protokollant bestimmt, der die wesentlichen Ergebnisse der Diskussion festhält.

Im Anschluss an die Fallstudie überarbeitet der Moderator die Aufzeichnungen des Protokollanten und stellt den Studierenden die wesentlichen Ergebnisse in einem Forum zur Verfügung. Diese Ergebnisse werden mit den Studierenden in einem abschließenden Chat abgestimmt und stellen somit einen Teil der Evaluation der Fallstudie dar. Neben dieser Form der Evaluation werden die Studierenden mittels eines Forums auch bei der Äußerung anonymer Kritik unterstützt.

Abschließend stellen die einzelnen Gruppen ihre Präsentationen bzw. Ergebnisse über die IULU auch den anderen Studierenden zu Verfügung. Die geforderte Verwendung von XML-Standards im Rahmen der Präsentationen sollte diesen Arbeitsschritt wesentlich vereinfachen. Mithilfe der IULU stehen die wesentlichen Ergebnisse der Fallstudie weiterhin zur Verfügung und können im Rahmen einer Nachbearbeitung und Prüfungsvorbereitung gezielt angewendet werden.

Tabelle 40 verdeutlicht tabellarisch die Ergebnisse auf der Entwicklungsebene ELM-E für die Fallstudie *Präsentieren von XML*.

Tabelle 40: ELM-E am Beispiel Fallstudie

| Phase | Zeitraumen | Präsentationsformat/ Materialien | Kommunikationsform |
|------------------------|---|--|---|
| Einführung | <ul style="list-style-type: none"> • 60 Minuten • Präsenzveranstaltung | <ul style="list-style-type: none"> • XML-Dokumente Repräsentation als Bildschirmpräsentation und/oder Skript • Videosequenzen | <ul style="list-style-type: none"> • Direkte Diskussion |
| Aktivität/ Interaktion | <ul style="list-style-type: none"> • 240 Minuten • Selbstlernphase: verteilt oder lokal | <ul style="list-style-type: none"> • XLink Linksammlung | <ul style="list-style-type: none"> • Diskussionsforum |
| Analyse/ Evaluation | <ul style="list-style-type: none"> • Präsenzveranstaltung | <ul style="list-style-type: none"> • Ergebnisse der einzelnen Gruppen als XML-Dokumente • Ergebnis der Abschlussdiskussion als Forumstruktur | <ul style="list-style-type: none"> • Direkte Diskussion • Chat zur abschließenden Diskussion • Forum zur anonymen Bewertung der IULU bzw. Fallstudie |

Evaluation

Die Inhalte der Lernumgebung wurden von den Studierenden nach Abschluss der ersten Durchführung durchgehend als relevant und qualitativ hochwertig beurteilt. Bezüglich der Auswahl der didaktischen Methode wurden sowohl in der Evaluation durch Fragebögen als auch in abschließenden Feedbackgesprächen durchgehend positive Ergebnisse erzielt. Diese müssen jedoch durch langfristige Evaluationsergebnisse weiter beobachtet werden. Da gerade in der universitären Ausbildung der Schwerpunkt noch immer auf Frontalunterricht liegt, ist durch die Verwendung anderer Methoden weiterhin die Möglichkeit gegeben, nur kurzfristiges positives Feedback durch den Hawthorne-Effekt (siehe 2.3.5.2) zu erhalten. Erst durch die langfristige Verwendung bestimmter Methoden und damit verbundenen Evaluationen ist der Erfolg dieser Konzeption zu beurteilen.

6.2.2 Lernumgebung *Grundlagen der Programmierung*

Im Folgenden wird die Entwicklung einer Lernumgebung *Grundlagen der Programmierung* beschrieben, die als internetbasierte Lernumgebung im Rahmen des Weiterbildungsstudienganges Wirtschaftsinformatik im Projekt *Virtuelle Aus- und Weiterbildung Wirtschaftsinformatik (VAWi)* [ABKP2001] entwickelt wurde. Zum Zeitpunkt der Veröffentlichung dieser Arbeit wird die Lernumgebung innerhalb einer Testgruppe im Weiterbildungsstudiengang evaluiert.

6.2.2.1 Anpassung

Die Entwicklungsprozesse in ELM wurden spezifisch für das Projekt VAWi angepasst. Dabei sind verschiedene Randbedingungen erfasst, die sich daraus ergeben, dass es sich um ein Verbundprojekt von mehreren Universitäten (Essen, Bamberg, Erlangen-Nürnberg) handelt und verschiedene Prozesse synchronisiert werden müssen. Die Ausführungen dieses Abschnitts beziehen sich auf die Entwicklungsarbeit der Universität Essen.

Zunächst werden die Prozesse des ELM, wie in Kapitel 5 beschrieben, vollständig durchlaufen. Hinzu kommen Erweiterungen, die sich aus der Projektstruktur ergeben. So werden verschiedene Prozesse unabhängig voneinander bearbeitet: Die Entwicklung der Lerninhalte erfolgt unabhängig durch die beteiligten Lehrstühle, während die Endprodukte (Lernumgebungen, Lernmaterialien) in einem gemeinsamen Qualitätssicherungsprozess evaluiert werden. Daher sind insbesondere die jeweiligen Konsensfeststellungen mit den Projektpartnern abzustimmen.

6.2.2.2 ELM-C

Kontextanalyse

Zunächst werden einerseits die *Projektziele* von VAWi, andererseits die Ausbildungsziele des Fachgebiets *Wirtschaftsinformatik der Produktionsunternehmen* erfasst. Ziel des Weiterbildungsstudienganges ist es, Studierenden unterschiedlicher Fachrichtungen eine fundierte theoretische und methodische Ausbildung zu ermöglichen und dabei Synergieeffekte, die sich aus den einzelnen Vorkenntnissen ergeben, zu nutzen. Es soll innerhalb der dreijährigen Projektlaufzeit ein Studiengang mit dem Umfang von 4.5 ECTS entwickelt werden. Der Kurs *Grundlagen der Programmierung* ist dabei Bestandteil eines Pflichtmoduls und ist als Einführungskurs vorgesehen.

Die IT-Analyse und die Ausbildungsanalyse entsprechen weitgehend den Ausführungen aus Abschnitt 6.2.1.2 und wird daher nicht nochmals erläutert.

Lernziele

Die Lernziele und eine grobe Strukturierung der Inhalte erfolgt zunächst für den kompletten Weiterbildungsstudiengang. Schwerpunkte wurden auf die Grundlagen der Systementwicklung, Organisation und Management sowie Basistechnologien gelegt. Im Wahlbereich werden einzelne Fachrichtungen der Wirtschaftsinformatik vertieft, um eine Spezialisierung zu ermöglichen. Das in den Wahlpflichtmodulen erlangte Methoden- und Theorienwissen soll anhand von praxisorientierten Aufgabenstellungen in Projektarbeiten vertieft werden. Eine Übersicht über die Inhalte zeigt Tabelle 41.

Tabelle 41: Inhalte von VAWi

| <i>Pflichtmodule</i> | |
|--|---|
| Organisation und Management (Betriebswirtschaftslehre) <ul style="list-style-type: none"> • Grundlagen der Betriebswirtschaftslehre • Rechnungswesen und Controlling • E-Business • Informations- und Wissensmanagement | Systementwicklung (Wirtschaftsinformatik) <ul style="list-style-type: none"> • Integrierte Anwendungssysteme, Datenbanken und Archivierungssysteme • Modellierung von Systemen und Prozessen |
| Basistechnologien (Informatik) <ul style="list-style-type: none"> • Rechner-, Betriebs- und Kommunikationssysteme, Verteilte Systeme • Grundlagen der Programmierung (Java) | Schlüsselqualifikationen I <ul style="list-style-type: none"> • Projektmanagement |
| <i>Wahlpflichtmodule/Projektarbeiten</i> | |
| Electronic Business <ul style="list-style-type: none"> • Enterprise Resource Planning • Supply Chain Management • E-Commerce • Projektarbeit | Entscheidungsunterstützung <ul style="list-style-type: none"> • Operations Research/Statistik • Simulation • Soft-Computing • Projektarbeit |
| Multimedia-Systeme <ul style="list-style-type: none"> • Multimedia-Technik • Hypermedia-Systeme • Mediendesign • Projektarbeit | Datenverwaltung <ul style="list-style-type: none"> • Datenmodellierung • Data Warehouse- und Data Mining- Systeme • Information Retrieval Systeme • Projektarbeit |
| Verteilte Systeme <ul style="list-style-type: none"> • Kommunikationssysteme und Netzwerkmanagement • Sicherheit in verteilten Systemen • Workflow- und Workgroup-Systeme • Projektarbeit | Electronic Learning <ul style="list-style-type: none"> • Lerntechnologien • Internetbasierte Lernumgebungen • Qualitätssicherung und Evaluation • Projektarbeit |
| Management der Systementwicklung <ul style="list-style-type: none"> • Methoden der Systementwicklung • Projekt- und Qualitätsmanagement • Entwicklungsumgebungen • Projektarbeit | Schlüsselqualifikationen II <ul style="list-style-type: none"> • Rechtliche Regelungen • Gesellschaftliche Entwicklungen • Geschäftsentwicklung • Projektarbeit |

Darauf folgend werden die Lernziele für den Kurs *Grundlagen der Programmierung* in ELM gesammelt, priorisiert und eine grobe Kursstruktur entworfen – die Sammlung und Bewertung der Lernziele erfolgt analog zu den Ausführungen in Abschnitt 6.2.1.2 und wird daher hier nicht nochmals vertieft. Eine Übersicht über die Lernziele ist in Abbildung 84 zusammengefasst.



Abbildung 84: Lernziele *Grundlagen der Programmierung*

Evaluation

Im Rahmen der Evaluation wurden allgemeine Qualitätskriterien bestimmt, die als Basis für eine interne Evaluation innerhalb des Projekts und eine externe Evaluation dienen. Dabei wurden folgende Qualitätskriterien in Anlehnung an [Kuge2001, Codo2001] zur Beurteilung einbezogen:

- *Handhabung der Lernumgebung*: Installation, Zugang, Navigation, Stabilität und Geschwindigkeit
- *Darstellung der Lernumgebung*: Aufbau, ästhetische Gestaltung, Lesbarkeit am Bildschirm, Inhalte, Steuerungselemente, Anweisungen, Strukturen der Masken, Schwierigkeitsgrad
- *Inhalte*: Relevanz (Theoriebezug, Methodenbezug, Praxisbezug), Aktualität, Erfassung relevanter Literatur und Quellen, Zielgruppenorientierung, Benutzeranpassung.
- *Didaktischer Aufbau*: Aufbaustruktur, Technologieeinsatz, Zusatzinformationen, Hilfefunktionen, Glossarverknüpfung, Begriffserklärungen, didaktischer Aufbau, Gliederung, Texte, Einstiegsniveau, Orientierung am Lernziel
- *Präsentation*: Grafiken, Animationen, Farbkombinationen, Grafikerklärungen, Visualisierungen, Realitätsnähe der Animationen, Bildschirmauflösung

- *Kommunikation*: Grad der Interaktivität, Kommunikationsanwendungen, Feedbackzyklen, verfügbare Anwendungen
- *Lernerfolg/Evaluation*: Fragestellungen, Gestaltung der Aufgaben, Rückmeldungen, Auswertung, Überprüfung des Lerninhalts, Korrigierbarkeit, Feedbackmöglichkeiten, Beurteilungsmechanismen

Neben der initialen Evaluation fließen die Ergebnisse aus der ersten Kursdurchführung im Rahmen von VAWi in die weitere Entwicklung ein. Dabei wird Feedback durch die Kommunikation in Foren bzw. durch direkte Befragung ermöglicht. Weiterhin stehen formale Kurzbefragungen zur Verfügung, um eine schnelle und anonyme Beurteilung einzubeziehen.

6.2.2.3 ELM-D

Wissensakquisition

In dieser Phase wurden zunächst relevante Literatur zur objektorientierten Programmierung und verfügbare Lernumgebungen recherchiert und erfasst. Dabei zeigte es sich, dass verschiedene generelle Ansätze vertreten werden. Einerseits wird unterschieden, ob bereits Programmiererfahrung besteht, andererseits unterscheidet sich der Aufbau der Lernmaterialien signifikant. Als Ergebnis dieser Phase wurde ein Überblick über aktuelle Literatur und den jeweiligen Beurteilungen spezifiziert, der dann als Basis für die weitere Entwicklung dient.

Benutzeranalyse

Die Benutzeranalyse der potenziellen Lernenden im Rahmen von VAWi ist wesentlich komplexer als im grundständigen Studium (siehe Abschnitt 6.2.1). Als einzige Zulassungsvoraussetzung wird ein Universitätsabschluss oder qualifizierter Fachhochschulabschluss gefordert. Somit setzt sich die Gruppe der Lernenden aus unterschiedlichen Fachrichtungen mit unterschiedlichem akademischen Hintergrund zusammen. Speziell bezogen auf das Erlernen objektorientierter Programmierung setzt sich die Gruppe zudem aus Personen mit hoher Programmiererfahrung (z. B. Programmierung umfangreicher Anwendungssoftware) und Anfängern zusammen. Dies hat ebenfalls Auswirkungen auf Motivation, Lerntempo und Lernpräferenzen der Studierenden.

Weiterhin handelt es sich um eine räumlich und zeitlich verteilte Studierendengruppe, was zu einer besonderen Bedeutung der asynchronen Kommunikation führt. Da sich die Gruppe sowohl aus Vollzeit- als auch aus Teilzeitstudierenden zusammensetzt, muss das Kommunikationsangebot flexibel gestaltet sein. Die Beschaffenheit der Lernervoraussetzungen und -präferenzen führt zu folgenden Richtlinien:

- *Individuelle Betreuung*: Um die unterschiedlichen Voraussetzungen ausgleichen zu können, wird ein hoher Betreuungsgrad vorgesehen.

- *Bildung heterogener Lerngruppen*: Synergieeffekte, die sich durch die unterschiedlichen Voraussetzungen ergeben, werden verstärkt genutzt. So werden Kleingruppen mit Lernenden mit unterschiedlichen Programmiererfahrungen gebildet, um gegenseitige Unterstützung innerhalb der Gruppe zu fördern. Durch diese Maßnahme werden zusätzlich Teamarbeitskompetenzen, die als maßgebliche soziale Lernziele erfasst wurden, erworben.
- *Kommunikation*: Es werden verschiedene synchrone wie asynchrone Kommunikationsanwendungen zur Verfügung gestellt, um zeitliche und räumliche Flexibilität zu erreichen.
- *Kontextsensitivität*: Aufgrund der heterogenen Studierendengruppe muss ein Bezug zum jeweiligen beruflichen oder privaten Kontext geschaffen werden. Dies erfolgt durch individuelle Betreuung oder Anpassung der Übungen.

Bezüglich der Lehrenden sind verschiedene Anforderungen gegeben. Neben der fachlichen Qualifikation muss eine didaktische Qualifikation im Bereich des Coachings über verschiedene Kommunikationsmedien vorhanden sein. Das bedeutet, dass Lehrende Inhalte und Methoden auch zur Laufzeit ändern und damit auf die Bedürfnisse der Lernenden eingehen können.

Methodenauswahl

Parallel zur Benutzeranalyse und den entsprechenden Maßnahmen wurden geeignete Methoden vom Entwicklungsteam in Zusammenarbeit mit den Studierenden evaluiert, die in einer räumlich und zeitlich verteilten Lernsituation einsetzbar sind. So betont [Schu2000] die Bedeutung des problemorientierten, kooperativen Lernens bei den komplexen Lerninhalten der objektorientierten Modellierung. Des Weiteren wird gefolgert, dass die Bearbeitung in einem realistischen, offenen Kontext notwendig ist und die Erarbeitung mithilfe entsprechender Werkzeuge erfolgen sollte. Ebenso hebt [Köll2001] die Nutzung von Werkzeugen hervor. Es zeigte sich, dass eine explorative Lernumgebung für das beschriebene Lernszenario geeignet ist (vgl. 2.3.8). So ist es den Lernenden möglich, das Lerntempo individuell anzupassen und Schwerpunkte innerhalb der Lerninhalte zu setzen.

Des Weiteren wurde festgelegt, dass aufgrund der Lernziele, der Lerninhalte und der Beschaffenheit der Lerngruppe ein hoher Anteil an kollaborativen Lernprozessen bestehen muss (z. B. im Bereich der Teamprogrammierung). Um den Gruppenbildungsprozess zu fördern, wurde ein Präsenzseminar mit umfangreichen Teambildungsprozessen vorgesehen. Diese Phase kann dabei nicht durch eine virtuelle Entsprechung ersetzt werden. Die Arbeit in Lerngruppen ist insbesondere bei der Bearbeitung von Übungen von Bedeutung, da gerade im Anwendungsfall besondere Anforderungen an Teamarbeitskompetenzen bestehen.

Zudem muss ein umfangreiches Betreuungsangebot durch synchrone (Chat und Videoconferencing) und asynchrone Kommunikation (E-Mail, Forum) zur Verfügung gestellt werden. Hier wird der Ansatz des Scaffolding (vgl. 5.5.2) verfolgt, so dass Studierende zunächst Problemlösungskompetenzen bei einem höheren Betreuungsgrad erlernen, während im weiteren Verlauf der Betreuungsaufwand gesenkt werden kann.

6.2.2.4 ELM-E

Ablauf festlegung/Lerneinheiten bearbeiten

Durch die Methode einer explorativen Lernumgebung ist das Lerntempo innerhalb der Lernmodule flexibel wählbar. Dennoch wird gerade für Anfänger, die noch keine Erfahrungen in der objektorientierten Modellierung und Programmierung haben, empfohlen, die ersten Lerneinheiten und Übungen (Einführung Objektorientierung) sequenziell zu bearbeiten. Die Übungen müssen zu einem feststehenden Termin eingereicht werden, um in den jeweiligen Lerngruppen einen ähnlichen Lernfortschritt zu erzielen und somit zu jeder Übung gegenseitige Unterstützung bzw. Hilfestellung zu fördern.

Dadurch ergibt sich die in Abbildung 86 gezeigte Navigation, die zudem die Layoutstandards innerhalb des VAWi-Projekts (Logo, Farbauswahl, Hervorhebung, Schriftarten) berücksichtigt.

The screenshot shows a web browser window displaying the course page for 'Grundlagen der Programmierung (Java)'. The page has a blue header with the VAWi logo and a navigation menu on the left. The main content area includes a table of lecturers, an abstract, and learning objectives.

| Prof. Dr. Helmut H. Adelberger (Lehrstiftung) | lewis@desi.uni-essen.de http://www.uni-essen.de Universitäts-Erzoffachbereich 5 (Wirtschaftswissenschaften) Wirtschaftsinformatik der Fachhochschule lehre@uni-essen.de http://www.uni-essen.de Universitäts-Erzoffachbereich 5 (Wirtschaftswissenschaften) Wirtschaftsinformatik der Fachhochschule Fakultät für Wirtschaftswissenschaften (Informatik) |
|---|--|
| Patrick Voith Dipl.-Wirt.-Inf. (Tutor) | |
| Zuordnung | |

Abstract

In diesem Kurs erwirbt der Student die Fähigkeit der objektorientierten Programmierung. Er ist in der Lage, Programmieranforderungen objektorientiert zu strukturieren und diese dann in JAVA umzusetzen. Insbesondere wird der Student in die Lage versetzt, mit Hilfe von application programming interfaces von Klassen selbst neue Klassen anzulegen. Dieser Kurs legt den Schwerpunkt auf objektorientierte Konzepte und versucht nicht, Java in all seinen Facetten zu vermitteln.

Lernziele

- Vermittlung der Begriffswelt der Programmierung
- Erarbeiten von Klassensystemen aus Case Studies
- Objektorientierter Programmieransatz mit UML
- Strukturiertes Programmieren
- Programmieren mit Java
- Kooperatives Lernen
- Kooperatives Programmieren

Methode und Ablauf

Abbildung 86: Navigation und Layout *Grundlagen der Programmierung*

Ein Schwerpunkt wird auf die individuelle Betreuung, Coaching und Scaffolding gelegt. Dies erfolgt in Foren unter Verwendung des Lernmanagementsystems ClearCampus [Clea2001]. Ein generelles Forum wird zur Verfügung gestellt, in dem eine Diskussion über allgemeine Handhabung, Struktur und technische Voraussetzungen des Kurses angeht. Des Weiteren werden Foren zu einzelnen inhaltlichen Schwerpunkten zur Verfügung gestellt, um einen strukturierten und nachvollziehbaren Kommunikationsverlauf zu

ermöglichen. Hinzu kommt eine individuelle Betreuung durch die Verwendung von Videoconferencing, Chat und E-Mail. Diese ist insbesondere für die Bearbeitung der Übungsaufgaben und Prüfungsaufgaben notwendig, um Verständnisprobleme zu beheben, Wissensunterschiede auszugleichen und individuelle Anpassungen vornehmen zu können.

Evaluation

Die Evaluation bezieht sich einerseits auf die Lernerfolgskontrolle, andererseits auf die interne und externe Evaluation der Lernumgebung.

Zur Lernerfolgskontrolle wird neben einer mündlichen Prüfung eine Übungsumgebung zur Verfügung gestellt, die die praktische Umsetzung erlernter Konzepte ermöglicht. Dabei wird zwischen Übungen zum Selbsttest und Übungen mit Prüfungsrelevanz unterschieden. Erste dienen als Möglichkeit, erlernte Inhalte praktisch umzusetzen, während die Prüfungsaufgaben als Teil der Bewertung gelten. Intern werden Beurteilungsfristen festgelegt, so dass die Zeit von der Einreichung einer Übung bis zum Feedback gering gehalten wird (maximal bis zu drei Tagen). Abbildung 87 zeigt ein Beispiel einer Übungsaufgabe.

Abbildung 87: Beispiel einer Prüfungsaufgabe

Wie bereits in Abschnitt 6.2.2.2 erläutert, wurden interne und externe Evaluationen durchgeführt, die sich jedoch auf Kursentwickler und externe Domänenexperten bezogen. Wichtigstes Evaluationsinstrument ist die Beurteilung des Kurses durch den Lernenden. So werden neben der Lernerfolgskontrolle informelle Befragungen im Forum durchgeführt, um kurzfristig auf Änderungsanforderungen und Schwierigkeiten einzugehen. Des Weiteren werden formelle, anonyme Befragungen durchgeführt, die ebenfalls zu Änderungen der Lernumgebung oder der Betreuung führen. Nach Abschluss der ersten Kursdurchführung werden diese Evaluationen detailliert analysiert und Änderungsanforderungen für ein Redesign des Kurses spezifiziert. Somit werden Inhalte, Methoden und damit verbundene Designentscheidungen stets auf die Bedürfnisse der Lernenden abgestimmt und auch zur Laufzeit einbezogen.

6.2.3 MobiLum: Prototyp einer mobilen Lernumgebung

Ziel von MobiLum (Mobile Lernumgebung) ist die Konzeption und prototypische Umsetzung einer Lernumgebung, die mobile Endgeräte, wie z. B. Mobiltelefone oder Personal Digital Assistants (PDA), einbezieht. Die Umsetzung dient als Basis für eine umfangreiche Untersuchung bezüglich der Eignung mobiler Lernumgebungen im universitären Kontext. Dabei gliedert sich die Entwicklung in zwei Teile:

- Es wird eine Lernumgebung zur Unterstützung der Präsenzveranstaltung *Computer Assisted Learning I* entwickelt, die per Desktop zur Vertiefung und Nachbearbeitung von Lerninhalten dient. Die *XML-based Explorative Learning Environment (XELE)* ist eine internetunterstützte, javascript-basierte Lernumgebung, die mit ELM erstellt wurde.
- Die Lernumgebung wird für den Einsatz mobiler Endgeräte angepasst. Dies bezieht sich einerseits auf eine angepasste Darstellung der XELE, andererseits auf die Schaffung didaktischer Szenarien, die den Einsatz eines mobilen Endgerätes sinnvoll erscheinen lassen. Eine Evaluation dieser Szenarien ist integraler Bestandteil der Entwicklung.

Dieser Abschnitt zeigt die Verwendung von ELM im Rahmen eines universitären Kontextes. Es wird gezeigt, wie die prototypische Umsetzung zur Evaluation neuer didaktischer Konzepte verwendet werden kann. Dabei wird deutlich, dass didaktische Konzepte gerade im Forschungsbereich des mobilen Lernens nicht ohne weiteres übernommen werden können. Durch die Verwendung von ELM wird eine prototypische Lernumgebung evaluiert und das didaktische Konzept so modifiziert, dass eine Erweiterung der Lernumgebung erfolgt. Des Weiteren werden Schlussfolgerungen bez. erfolgversprechender didaktischer Konzeptionen des mobilen Lernens gezogen, die aus der Evaluation abgeleitet werden können.

6.2.3.1 ELM-C

Projektinitiierung/Kontextanalyse

Die Anpassung des Essener-Lern-Modells erfolgt in dieser Entwicklung analog zur in 6.2.1.1 beschriebenen Anpassung. Ebenso entspricht der Entwicklungskontext (universitäre Veranstaltung im Rahmen des Studiums der Wirtschaftsinformatik) dem aus 6.2.1, daher wird in diesem Abschnitt nur auf Besonderheiten dieses Entwicklungsprozesses eingegangen.

Die Analyse der IT-Struktur entspricht ebenfalls der aus 6.2.1, dennoch muss ergänzend eine Analyse der technischen Anforderungen in Bezug auf mobile Endgeräte erfolgen.

Curriculumanalyse/Evaluation

Die Anpassung der ELM-Applikation erfolgt spezifisch für die Entwicklung von Lerneinheiten im Fachgebiet *Wirtschaftsinformatik der Produktionsunternehmen (WiP)* der Universität Essen. Die Entwicklung der Lernumgebung XELE erfolgte für eine bereits bestehende Veranstaltung des Hauptstudiums der Wirtschaftsinformatik. Während der Lernziel-sammlung wurde das bisherige Vorlesungsprogramm evaluiert, überarbeitet und folgende Schwerpunkte festgelegt:

- *Sichtweisen auf Lernprozesse*: Die Interdisziplinarität des Forschungsgebiets soll analysiert werden. Psychologische, pädagogische und IT-Aspekte werden diskutiert.
- *Didaktische Grundlagen* enthalten die didaktische Konzepte im umfassenden Sinne. So werden Planungsmodelle wie auch didaktische Methoden diskutiert.
- Die *Klassifikation von Lernumgebungen* dient der Übersicht über verschiedene methodische Ansätze zur Entwicklung von Lernumgebungen.
- Die *Architektur von Lernumgebungen* liefert eine Übersicht über die Komponenten von Lernumgebungen, die dann im Detail untersucht werden: Wissensbasis, Methodenbasis, Benutzermodell, Präsentations-, Kommunikations- und Evaluationskomponente.
- Der Schwerpunkt bei der *Entwicklung von Lernumgebungen* liegt auf der praktischen Anwendung der zuvor behandelten theoretischen Konzepte. Es werden verschiedene Entwicklungswerkzeuge eingesetzt, die den Entwicklungsprozess von der Konzeption bis zur Implementierung unterstützen.

Während der Evaluationsphase wurden die neuen Lernziele, die gegenüber dem ursprünglichen Kurs wesentlich modifiziert wurden, diskutiert. Neben Autor und Entwickler wurden Studierende einbezogen, die bereits die Präsenzveranstaltung besucht hatten und ihre Erfahrungen einbringen konnten. Dabei wurden die einzelnen Lernziele insbesondere auf Aktualität überprüft und entsprechend erneuert. Gerade im Bereich der Architekturen und einzelnen technologischen Komponenten zeigte sich, dass eine solche Überarbeitung in regelmäßigen Zyklen von besonderer Bedeutung ist, um die inhaltliche Qualität sicherzustellen. Abbildung 88 zeigt das Modell der Inhalte in ELM.

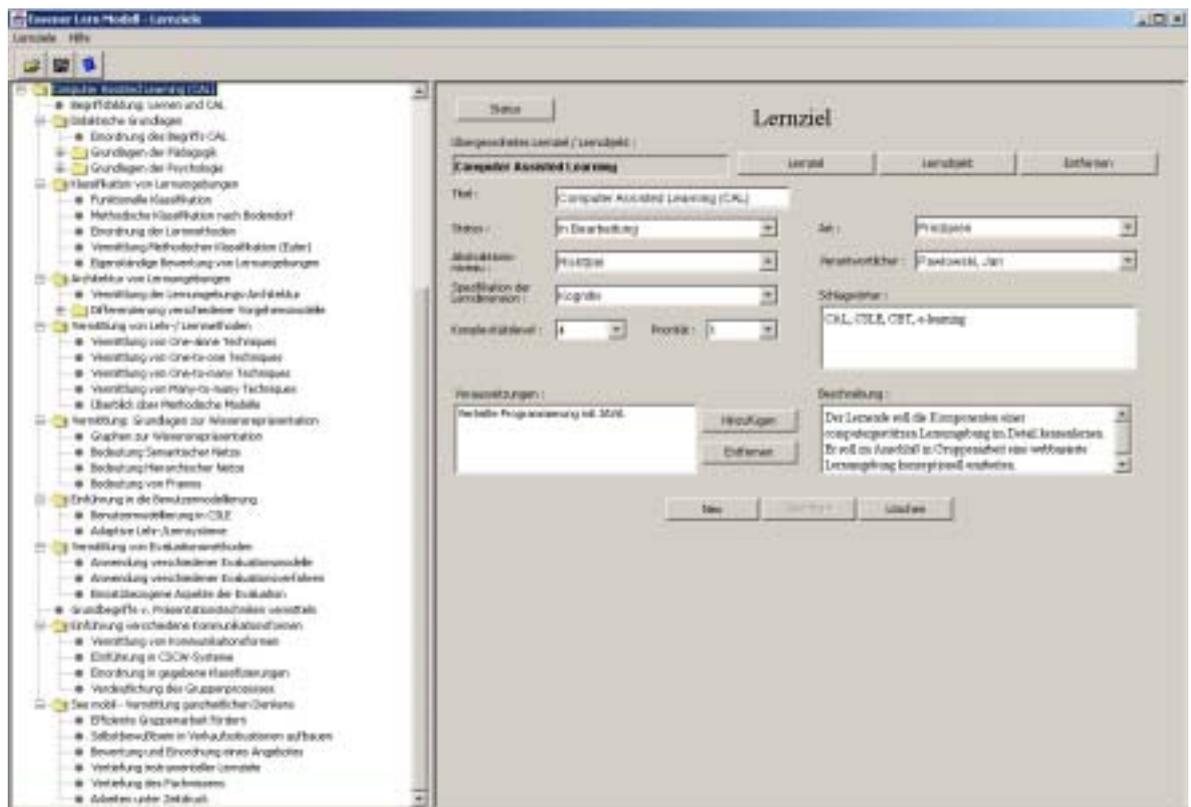


Abbildung 88: Lernziele XELE

6.2.3.2 ELM-D

Wissensakquisition

Im Rahmen der Wissensakquisition werden zum einen neue Ressourcen im Anwendungskontext recherchiert, zum anderen werden die bereits bestehenden Inhalte der Vorlesung, die in Form von Microsoft Powerpoint-Präsentationen und entsprechender Literatur vorliegen, geprüft. Während der Akquisitionsphase zeigte es sich, dass nicht alle bestehenden Materialien ohne weiteres in die neue Lernumgebung übernommen werden konnten und daher neu entwickelt werden mussten. Gerade die Informationsdarstellung in mobilen Lerneinheiten erforderte eine neue Repräsentation und inhaltliche Aufbereitung.

Nutzeranalyse

Das Ergebnis der entsprechenden Nutzeranalyse entspricht der Analyse aus 6.2.1.3. Zudem wurde bereits im Vorfeld bei Teilnehmern recherchiert, welche technischen Voraussetzungen gegeben waren. Es zeigte sich, dass PDA oder Organizer mit Internetzugang derzeit noch wenig verbreitet sind. Dagegen waren WAP-fähige (Wireless Application Protocol) Mobiltelefone größtenteils vorhanden. Dennoch werden derzeit nur Übertragungsverfahren

der zweiten Generation (z. B. GSM, Globales System für Mobilkommunikation) genutzt, die nur geringe Datenübertragungsraten zulassen.

Methodenauswahl

Die Lernumgebung basiert auf drei unterschiedlichen Methoden, die das didaktische Szenario beschreiben:

1. *Präsenzanteil*: In dieser Phase werden die Lerninhalte des Kurses durch verschiedene Methoden erarbeitet. Neben klassischen Vorlesungen werden Inhalte (z. B. verschiedene Klassen von Lernumgebungen) von Studierenden eigenständig erarbeitet. Der Lehrende übernimmt eine Coaching-Funktion. So werden via Internet relevante Inhalte erarbeitet, analysiert und beurteilt.
2. *Explorative Lernumgebung*: In der explorativen Lernumgebung haben Studierende die Möglichkeit, Inhalte eigenständig zu erarbeiten oder zu vertiefen. Dabei sind die Inhalte in den Anwendungskontext (Entwicklung von Lernumgebungen) eingebunden, der im weiteren Verlauf Inhalt des Planspiels ist. Dies dient als Vorbereitung für das Planspiel. Die explorative Lernumgebung kann einerseits via Desktop-PC zugegriffen werden, andererseits werden Ausschnitte (siehe 6.2.3.3) mobil zugänglich gemacht.
3. *Mobiles Planspiel*: Das Planspiel wird in der Präsenzphase begonnen und dann mobil durchgeführt. Die Gruppenarbeit kann unter Verwendung der Kommunikationsapplikationen (Chat, E-Mail, Foren, Application Sharing durch Microsoft Netmeeting) erfolgen. Neue Problemstellungen werden vor den einzelnen Aktivitätsphasen einerseits als Website, andererseits als WAP-Seite zur Verfügung gestellt.

Das mobile Planspiel besteht aus folgenden Phasen, die in ELM modelliert wurden und somit auch für andere Inhalte wiederverwendbar sind (siehe Abbildung 89). So wurden Phasen bzw. Phasenblöcke wiederverwendet, die bereits im Kontext von Abschnitt 6.2.1.3 erläutert wurden:

- *Einführung*: In dieser Phase, die während der Präsenzveranstaltung erfolgt, werden zunächst die Regeln und die Problemstellung des Planspiels erläutert. Darauf folgend werden Kleingruppen zu je drei bis fünf Personen gebildet, die über eine differierende Vorbildung verfügen (BWL bzw. Wirtschaftsinformatik). Die Aufgabenstellung, die sich auf die Entwicklung einer Lernumgebung zu verschiedenen Themen bezieht, ist außerdem als WWW-Seite verfügbar. Die Themen werden je nach Vorwissen zur Laufzeit des Planspieles vom Moderator angepasst.
- *Aktivität*: In einer räumlich verteilten Phase erfolgt die erste Aktivität. In dieser Phase wird ein Grobkonzept erstellt, das als WWW-Seite umgesetzt werden soll. Des Weiteren werden mehrfach Änderungsanforderungen via Messaging an die Mobiltelefone der Teilnehmer versendet.
- *Reflexion/Abstraktion*: Die Reflexionsphase besteht aus der Beurteilung der Grobkonzepte durch die Moderatoren via Chat und Application Sharing. Dabei werden die Lösungen der einzelnen Gruppen synchron kommentiert und Hilfestellungen gegeben.

Zudem erfolgt ein kurzes Feedback bezüglich der mobilen Änderungsanforderungen (Erhalt, Einfluss auf den Lernprozess, Motivation).

- *Aktivität*: In einer zweiten Aktivitätsphase werden Feinkonzepte erstellt, die wiederum beurteilt werden (via Forum). Die Beurteilung muss dabei kurzfristig (innerhalb eines Tages) erfolgen, um den Lernprozess nicht zu unterbrechen. Änderungsanforderungen werden mobil durch Messaging mitgeteilt. Diese Änderungsanforderungen werden als Phase der *außerordentlichen Aktivität* zusammengefasst. Diese Phase erfordert eine sofortige Reaktion auf Ereignisse und soll somit zu zusätzlicher Motivation im Spiel führen.
- *Analyse/Evaluation*: Die Abschlussbeurteilung und Zusammenfassung erfolgt innerhalb einer Woche in einem Forum. Dazu stehen Beurteilungsfragebögen zur Verfügung. Außerdem erfolgt eine Diskussion über den Erfolg des Planspiels.

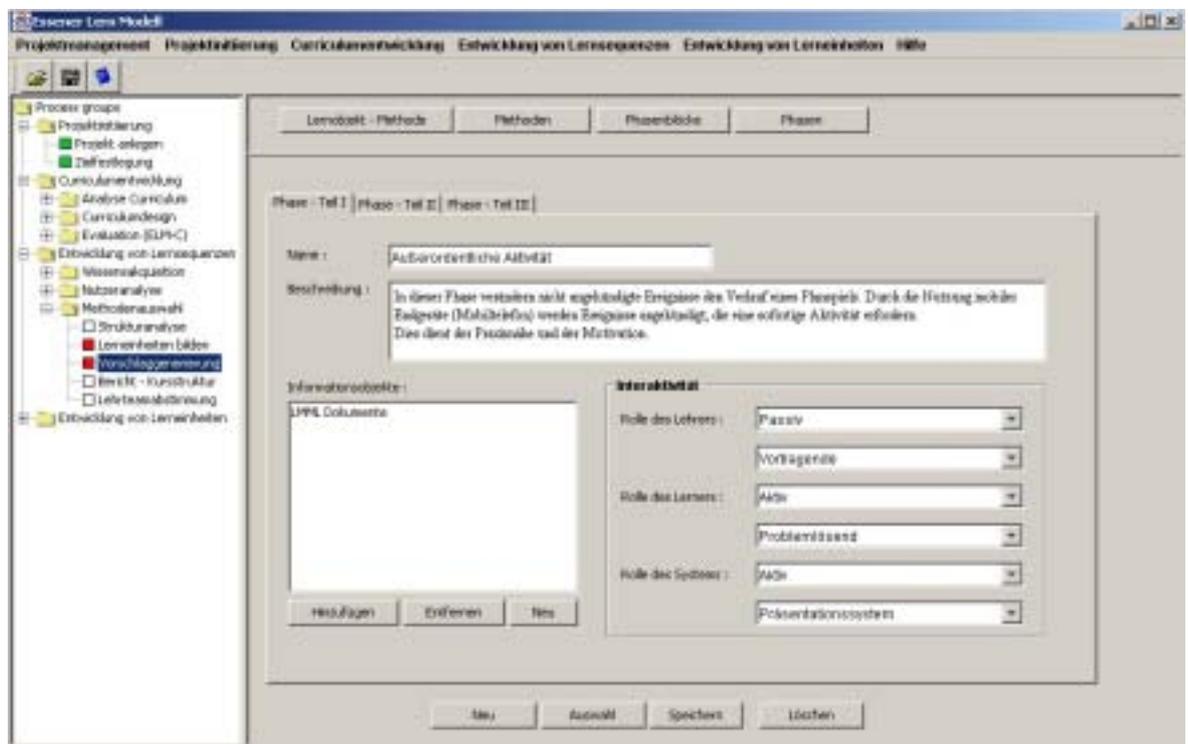


Abbildung 89: Methodenmodell des mobilen Planspiels

Design der Lernumgebung

Das Design bezieht sich insbesondere auf die Lerninhalte des Kurses. Abbildung 90 fasst die Lernobjekte der Lernumgebung zusammen.

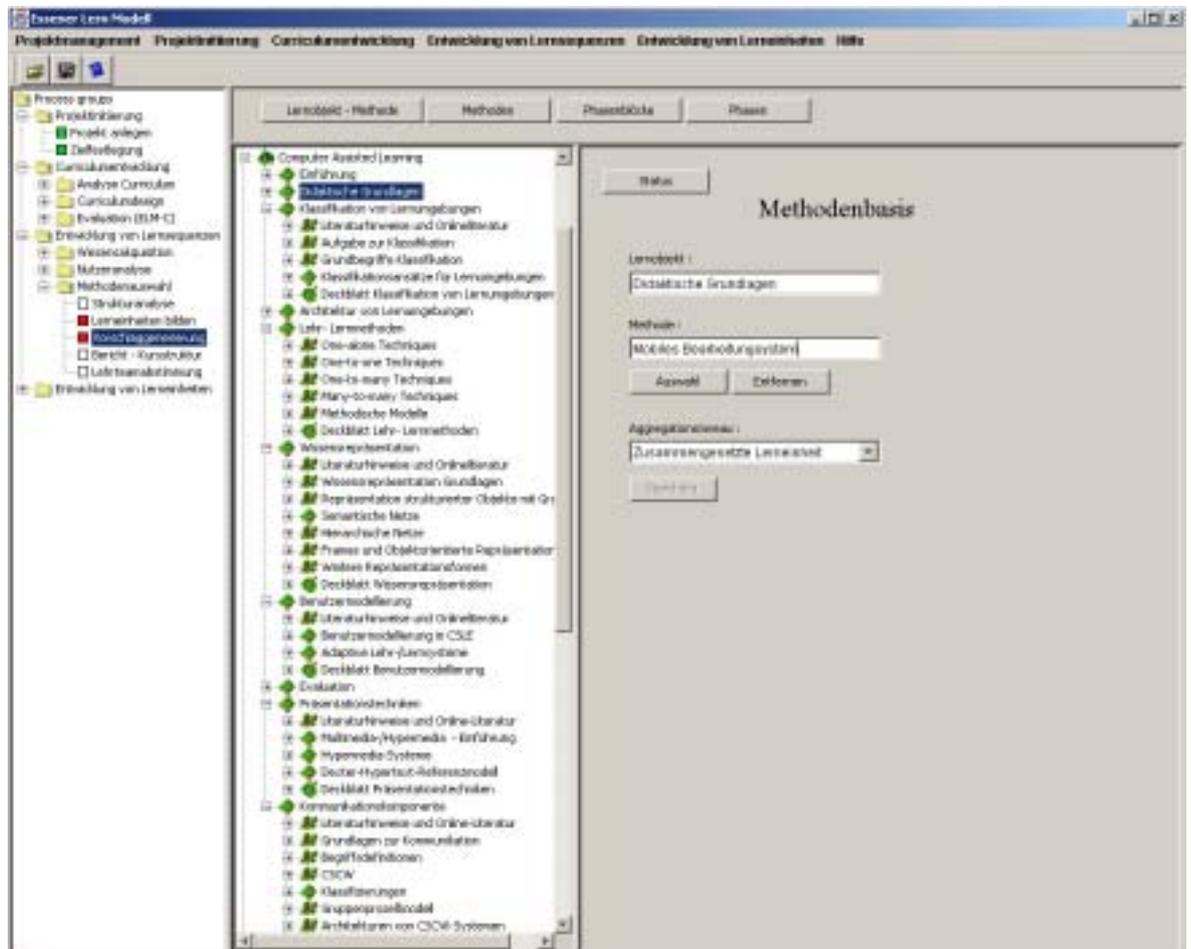


Abbildung 90: Lernobjekte XELE

6.2.3.3 ELM-E

Ablauflegung/Lerninhalte bearbeiten

Der Kurs wird wöchentlich als Präsenzveranstaltung durchgeführt und ist durch die Semestertermine beschränkt. Alle Lerninhalte stehen in der explorativen Lernumgebung durchgängig zur Verfügung. Das mobile Planspiel wird in einem Zeitraum von zwei Wochen am Semesterende durchgeführt, da zu diesem Zeitpunkt die zur Lösung notwendigen Lerninhalte bereits bearbeitet wurden.

Die Lerninhalte wurden unter Verwendung von LMML überarbeitet und als IULU zur Verfügung gestellt. Dabei werden in der mobilen Version nur Elemente angezeigt, die als Zusammenfassung gekennzeichnet sind. Dadurch ergibt sich die Möglichkeit, zu jedem Zeitpunkt Inhalte nachzubearbeiten und zu vertiefen. Die Bereitstellung der kompletten Lernumgebung oder umfangreiche graphische Elemente werden nicht als sinnvoll erachtet, da die Übertragungsraten zu gering sind. Ebenfalls wurde diese Möglichkeit von den Studierenden als nicht sinnvoll angesehen.

Präsentation/Kommunikation

Die Methode der explorativen Lernumgebung ist mit einer Navigationsstruktur verknüpft, die sich an der Hierarchie und den Verknüpfungen der Lernobjekte orientiert. Somit kann innerhalb der Kurshierarchie jede Lerneinheit direkt aufgerufen werden. Eine zweite Möglichkeit zur Navigation in der Lernumgebung ist die Nutzung der Navigations-Buttons, die am unteren Rand des Bildschirms platziert sind. Sie geben dem Lernenden die Möglichkeit, die Lerneinheiten sequenziell, im Sinne eines Leitfadens, durchzuarbeiten. Darüber hinaus wird die Überschrift der Lernsequenz, die der Lernende gerade bearbeitet, am oberen Bildschirm dargestellt, um das Phänomen des *Lost in Hyperspace* [Conk1987] zu verhindern. Hinzu kommt die Chatfunktion in einem weiteren Fenster.

Das Bildschirm-Layout, das in der Lernumgebung verwendet wird, ist für den Einsatz auf Desktop PCs optimiert und somit zunächst nicht für mobile Endgeräte geeignet. Um den Bildschirmauflösungen von PDA gerecht zu werden, wird das Layout für MobiLum in folgender Weise verändert (vgl. Abbildung 91):

Die Navigation über die Kursstruktur, die einen nicht unerheblichen Teil des Bildschirms ausfüllt, wird in ein separates Fenster ausgelagert. Die drei Bereiche für die Überschrift, den Lerninhalt und die Navigations-Buttons nehmen nun den gesamten Bildschirm ein. Bei der Arbeit mit der Lernumgebung tritt das Fenster mit der hierarchischen Navigationsstruktur in den Hintergrund und kann vom Lernenden bei Bedarf angewählt werden, um andere Lernsequenzen aufrufen zu können.

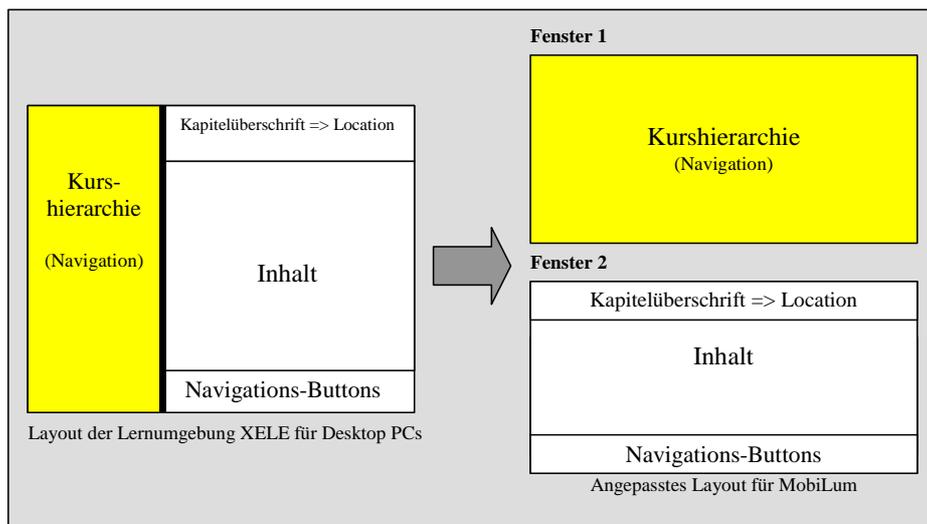


Abbildung 91: Schematische Übersicht der Bildschirmgestaltung

[Strz1997] erachtet es als wichtig, dass der Lernende die Möglichkeit zur Artikulation und Selbstreflexion über eigene Lern- und Lösungsstrategien hat. Diese Phase wird in vielen Lernumgebungen durch Chats oder Foren unterstützt. MobiLum integriert erstmals Messaging-Funktionen zur Kommunikation mit anderen Lernenden. Wichtig ist weiterhin die tutorielle Hilfe durch den Kursbetreuer, die den Studierenden in Form von fest terminierten synchronen Chats bzw. asynchronen Foren geboten wird (Abbildung 92).

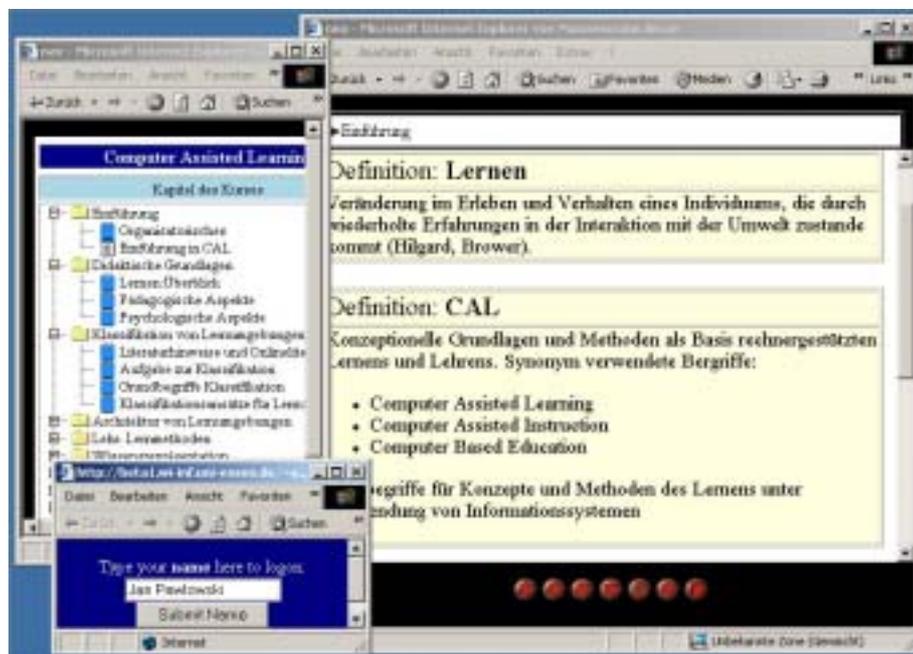


Abbildung 92: Layout MobiLum

Die Systemarchitektur von MobiLum, die auf XELE aufsetzt, kann exemplarisch durch den folgenden Ablauf beschrieben werden (Abbildung 93):

- *Erstellung eines Benutzerprofils:* In diesem Schritt wird ein Benutzerprofil erstellt (1), so dass ein individuelles Stylesheet (2) erzeugt wird.
- *Generierung der Lernumgebung:* Bei der Anforderung einer Ressource (3) wird unter Verwendung des Stylesheets und durch die Anforderung der entsprechenden Lernobjekte aus dem Repository (4) eine Lernumgebung generiert, die dann an den Lernenden gesendet wird (5). Wird eine mobile Lernumgebung generiert, so werden nur LMML-Zusammenfassungen erzeugt, im anderen Fall werden die Elemente erzeugt, die vom Benutzer im Benutzerprofil festgelegt wurden (Definitionen, Beispiele).

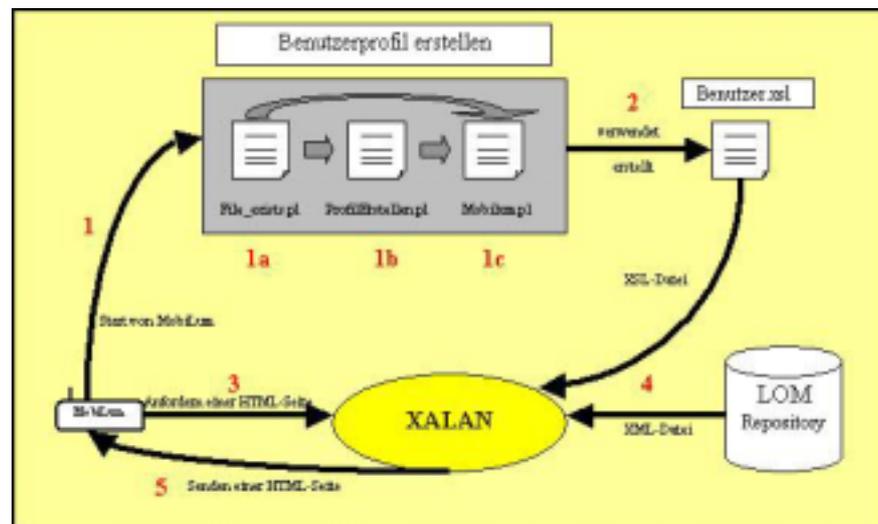


Abbildung 93: MobiLum-Architektur

Evaluation

Der Kurs wurde zunächst zu Forschungszwecken entwickelt und wird erstmals Anfang 2002 eingesetzt. Die Evaluationsplanung umfasst die Evaluation der Lernumgebung und des Planspiels, um den Erfolg der didaktischen Konzeption, insbesondere der mobilen Komponenten zu überprüfen. Dabei wird das Bewertungsschema aus 6.2.2.4 übernommen.

Als Hypothese wird aufgrund bestehender Erfahrungen [vgl. R ath2001, Noki2001] erwartet, dass der Einsatz von Mobiltelefonen der 2. Generation derzeit nur f ur die Verwendung der Kommunikationsfunktionen sinnvoll ist. SMS-Dienste sind bereits weit verbreitet und akzeptiert und werden daher h ufig genutzt. Kurze textuelle Pr sentationen erscheinen ebenfalls sinnvoll nutzbar, jedoch sind diese eher f ur die Verwendung durch PDA geeignet. Durch die zuk unftige Einf uhrung von Endger aten der dritten Generation (z. B. UMTS, Universal Mobile Telecommunication System) ist allerdings eine wesentliche  nderung des Kommunikationsverhaltens und der Darstellungsm glichkeiten zu erwarten. ELM wird dahingehend weiter entwickelt, dass mobile Kommunikations- und Informationsobjekte integriert und neue didaktische Szenarien entwickelt und evaluiert werden k nnen.

7 Zusammenfassung und Ausblick

7.1 Zusammenfassung

In diesem Abschnitt werden die wesentlichen Ergebnisse dieser Arbeit zusammengefasst.

Entwicklung eines generischen, flexiblen und adaptierbaren Vorgehensmodells

Bestehende Vorgehensmodelle wurden auf Ihre Eignung zur Entwicklung computerunterstützter Lernumgebungen überprüft. Dabei wurden verschiedene Vorgehensmodelle aus unterschiedlichen Anwendungskontexten analysiert und bewertet. Das Essener-Lern-Modell integriert die positiven Eigenschaften der analysierten Modelle und bildet somit eine Weiterentwicklung für den Kontext computerunterstützter Lernumgebungen. Es beschränkt die potenziellen Nutzer und Anwender dabei nicht in ihrer Gestaltungsfreiheit. So kann das Vorgehensmodell individuell modifiziert und auf die Bedürfnisse der Anwender angepasst werden. Die Erweiterbarkeit erleichtert das Einfügen neuer Modelle und Methoden.

Beschreibung einer Architektur für computerunterstützte Lernumgebungen

In dieser Arbeit wurde eine Architektur als Rahmen zur Einordnung, Beschreibung und zum Vergleich von Lernumgebungen entwickelt. Die Architektur des Essener-Lern-Modells und die korrespondierenden Prozesse dienen als Basis für die Entwicklung interoperabler Lernumgebungen.

Integration von Lernprozessen und Geschäftsprozessen

Das Essener-Lern-Modell wurde unter Verwendung des ARIS-Toolset prozessorientiert modelliert, so dass die Prozesse zur Erstellung von Lernumgebungen in transparenter Form vorliegen und in bestehende Prozessmodelle integriert werden können. Somit ist bei der Anwendung des Modells abschätzbar, welche Ressourcen zur Erlangung bestimmter Kompetenzen bzw. Lernziele benötigt werden. Durch die Verbindung von Ausbildungs- und Geschäftsprozessen werden betriebliche Abläufe wesentlich transparenter gestaltet. Des Weiteren erleichtert dieses Vorgehen die Integration bestehender Informationssysteme und die Verbindung von Ausbildung und täglichen Arbeitsabläufen. Als Beispiel können realistische Szenarien als Anwendungsbeispiele oder Fallstudien von bestehenden Informationssystemen in Lernumgebungen übernommen werden. Dadurch wird die klassische Trennung von Arbeit und Lernen aufgehoben.

Durchgehende Unterstützung des Entwicklungsprozesses von der Curriculumentwicklung bis zur Umsetzung einzelner Lernumgebungen

Verschiedene Vorgehensmodelle aus Softwareentwicklung, Didaktik und Wirtschaftsinformatik wurden eingehend analysiert. Es zeigte sich, dass mithilfe solcher Modelle der Schwerpunkt nur auf die Entwicklung einzelner Komponenten von Lernumgebungen (z. B. Hypermedia-Systeme, Medieneinsatz) oder auf bestimmte Klassen von Lernumgebungen (z. B. CBT, Selbstlernumgebungen) gelegt werden kann. Das vorliegende generische Modell umfasst ein breites Einsatzfeld, indem die Curriculumentwicklung, die Entwicklung von Lernumgebungen, die Entwicklung von Lerneinheiten und umfangreiche Projektmanagementfunktionen unterstützt werden.

Integration und Nutzung von Lerntechnologiestandards

Das Essener-Lern-Modell unterstützt die Spezifikation und Implementierung von Lerntechnologiestandards. Die Spezifikation dieser Standards (wie z. B. Learning Object Metadata) ist integraler Bestandteil Teil des Entwicklungsprozesses. Um den zusätzlichen Spezifikationsaufwand zu minimieren, wurden Methoden entwickelt, mit denen Metadaten teilweise automatisch generiert werden können (z. B. Ableitung aus anderen Projektspezifikationen). Weiterhin werden Metadaten nicht nur zur Beschreibung, sondern auch bei der Erzeugung von Lernumgebungen verwendet (z. B. in der Generierung der Navigationsstruktur oder von Zusammenfassungen). Des Weiteren wurde ein Kontext- und Methodenmodell entwickelt, das die Schwachstellen der existierenden Standards behebt: In Verbindung mit bestehenden Standards wurde somit ein Gesamtkonzept für interoperable Lerntechnologien auf der Basis von Standards entwickelt.

Modellierung didaktischer Methoden

Es wurde gezeigt, dass verschiedene Ansätze zur Modellierung didaktischer Methoden existieren, diese jedoch nur schwer vergleichbar sind. Durch die entwickelte Spezifikation eines Methodenmodells wird ein Ansatz zur Standardisierung der Darstellung didaktischer Methoden vorgelegt.

Plattformunabhängige und interoperable Einsetzbarkeit der ELM-Applikation

Die Implementierung des Essener-Lern-Modells, die ELM-Applikation, kann sowohl internetbasiert als auch in Einzelplatzkonfigurationen eingesetzt werden. Die Ergebnisse der Konzeption (z. B. Lernzielnetzwerk, Standardspezifikationen, Lernmodule, Zeitpläne) werden in der Extensible Markup Language (XML) generiert und sind somit von anderen Applikationen nutzbar und integrierbar. Dieses Tool wird im Fachgebiet *Wirtschaftsinformatik der Produktionsunternehmen* und im Projekt *VAWi* zur Planung und Umsetzung von Lernumgebungen eingesetzt und weiterentwickelt.

Entwicklung XML-basierter Lernumgebungen

Mithilfe des Essener-Lern-Modell wurden bereits verschiedene Kursentwicklungen durchgeführt. Der Entwicklungsprozess wurde am Beispiel der Entwicklung einer Lernumgebung *XML* im Fachgebiet Wirtschaftsinformatik der Produktionsunternehmen, einem Kurs *Grundlagen der Programmierung* im Projekt VAWi und der mobilen Lernumgebung *MobiLum* gezeigt. Diese Entwicklungen dienen als Basis für zukünftige Evaluationen und als Leitfaden für weitere Entwicklungen.

7.2 Ausblick

Neben den Ergebnissen lassen sich aus der vorliegenden Arbeit neue Forschungspotenziale ableiten, die eine weitere Arbeit in dem Themenbereich notwendig machen. Dabei sind besonders in folgenden Bereichen weitere Forschungen notwendig:

Electronic Education Markets

Neben Entwicklungen der Virtualisierung von Bildungseinrichtungen werden zunehmend Weiterbildungsangebote auf elektronischen Märkten verfügbar gemacht [vgl. HäMV1996, Hämä1999, LaBo2000, KräM1999]. Dabei bezeichnet ein *Electronic Education Market (EEM)* ein interorganisationales Informations- und Kommunikationssystem, das die Aufgaben von Administrations-, Entwicklungs- und Lernsystemen ganz oder teilweise übernimmt und zusätzlich betriebswirtschaftliche Handelstransaktionen unterstützt. Die Verwendung von Standards wie SCORM, LOM, PAPI oder CMI ist dabei unerlässlich [PaAd2001b]. Lernumgebungen, die mit ELM entwickelt werden, können ohne zusätzlichen Aufwand auf elektronischen Märkten eingesetzt werden. Dennoch muss gerade die Anbindung betriebswirtschaftlicher Informationssysteme und korrespondierender Handelsprozesse noch weiter untersucht werden. Eine detaillierte Ausführung unter Einbeziehung von Electronic Business-Standards (vgl. 4.2.1.2) findet sich in [PaAd2001b].

Standardisierung

Der Bereich der Standardisierung von Lerntechnologien ist derzeit noch in der Entwicklung. Die entwickelten Standards werden ständig überarbeitet und sind noch nicht als Norm (z. B. ISO) verabschiedet. Daher müssen neue Entwicklungen (z. B. SCORM, Version 1.2) in das Essener-Lern-Modell integriert werden. Gerade im Bereich der Standardisierung der Darstellung didaktischer Methoden kann das vorliegende Modell einen wesentlichen Beitrag zur Standardisierungsdiskussion liefern.

Weiterentwicklung des Essener-Lern-Modells

Die ELM-Applikation wird kontinuierlich weiterentwickelt. So sollen zum Beispiel die modellierten didaktischen Methoden erweitert werden, um eine umfangreiche Methoden-

basis zur Verfügung zu stellen und damit Entwicklern beim Design methodisch vielfältiger Lernumgebungen zu unterstützen. Zudem werden aus durchgeführten Kursen Erfahrungen gewonnen und in der Methodenbasis abgebildet, so dass die Vorschlaggenerierung stetig verbessert wird.

Weiterentwicklung ELM-Run-Time

Die Run-Time-Komponente des Essener-Lern-Modells wurde in der vorliegenden Arbeit zu Evaluationszwecken implementiert. Die Funktionen dieser Komponente sollen so erweitert werden, dass ein SCORM-konformes LMS entsteht. Dieses wird insbesondere zur Entwicklung und Evaluation von Standards dienen.

Integration mobiler Technologien und Methoden

Durch die Verfügbarkeit mobiler Technologien werden auch mobile Lernumgebungen wachsende Bedeutung gewinnen. Dabei ist zu erwarten, dass technologieorientierte Ansätze bei den ersten Entwicklungen vorherrschen werden. Um auch in diesem Bereich adäquate Lernsituationen aus didaktischer wie technologischer Sicht zu schaffen, werden mobile Anwendungen und mögliche didaktische Methoden ins Essener-Lern-Modell integriert und evaluiert.

Literaturverzeichnis

- [ABKP1999] *Adelsberger, H.H., Bick, M.H., Kraus, U.F., Pawlowski, J.M.*: A Simulation Game Approach for Efficient Education in Enterprise Resource Planning Systems. In: Proc. of ESM 99 - Modeling & Simulation: a Tool for the next Millennium. Warsaw 1999.
- [ABKP2001] *Adelsberger, H.H., Bick, M., Körner, F., Pawlowski, J.M.*: Virtual Education in Business Information Systems (VAWi) - Facilitating collaborative development processes using the Essen Learning Model. ICDE World Conference, Düsseldorf 2001.
- [AdBP2000a] *Adelsberger, H.H., Bick, M.H., Pawlowski, J.M.*: The Essen Learning Model - A Step Towards a Standard Model of Learning Processes. In: *Bordeau, J., Heller, R. (Hrsg.)*: Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2000.
- [AdBP2000b] *Adelsberger, H.H., Bick, M.H., Pawlowski, J.M.*: Design Principles for Teaching Simulation with Explorative Learning Environments. In: Proc. of Winter Simulation 2000. Orlando, FL, USA 2000.
- [AdCP2001] *Adelsberger, H.H., Collis, B., Pawlowski, J.M. (Hrsg.)*: Handbook on Information Systems for Education and Training. In: *Bernus, P., Blazewicz, J., Schmidt, G., Shaw, M. (Hrsg.)*: International Handbook on Information Systems. Springer, Hamburg 2001.
- [AdKo1995] *Adelsberger, H.H., Körner, F.*: Data Modeling with IDEF1X. In: *Adelsberger, Lazansky, Marik (Eds.)*: Information Management in Computer Integrated Manufacturing. Lecture Notes in Computer Science 973, Springer, Berlin 1995.
- [AdKP1998a] *Adelsberger, H.H., Körner, F., Pawlowski, J.M.*: A Conceptual Model for an Integrated Design of Computer Supported Learning Environments and Workflow Management Systems. In: Proc. of Teleteaching '98: Distance Education, Training and Education, XV. IFIP World Computer Congress, IFIP, Vienna 1998.
- [AdKP1998b] *Adelsberger, H.H., Körner, F., Pawlowski, J.M.*: Computerunterstütztes Lernen als integrierter Bestandteil von Workflow Management Systemen. Frühjahrstagung Wirtschaftsinformatik, Hamburg 1998.
- [AdKP1999] *Adelsberger, H.H., Körner, F., Pawlowski, J.M.*: Continuous Improvement of Workflow Models Using an Explorative Learning Environment. In: Proc. of ED-MEDIA 1999, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Seattle, WA 1999.
- [Aeb1976] *Aebli, H.*: Grundformen des Lernens. 9. Aufl., Klett, Stuttgart 1976.
- [Ahro1998] *Ahronheim, J. R.*: Descriptive metadata: Emerging standards. In: Journal of Academic Librarianship 24 (1998) 5.
- [Akin1988] *Akin, M.C.*: Quality of Educational Indicators. In: Studies in Educational Evaluation 14 (1988).
- [AnCV1999] *Anderson, T., Collier, G., Veres W.*: IMS Enterprise Best Practice and Implementation Guide, Version 1.01. <http://www.imsproject.org/enterprise/enbest03.html>, 1999-12-21, Abruf am 2000-08-12.
- [Ande2000a] *Anderson, T.*: IMS Content Packaging Information Model, Version 1.0. <http://www.imsproject.org/content/packaging/cpinfo10.html>, 2000-06-02, Abruf am 2000-08-12.
- [Ande2000b] *Anderson, T.*: IMS Content Packaging XML Binding, Version 1.0. <http://www.imsproject.org/content/packaging/cpbind10.html>, 2000-06-02, Abruf am 2000-08-12.

- [Ande2000c] *Anderson, T.*: IMS Content Packaging Best Practice & Implementation Guide, Version 1.0. <http://www.imsproject.org/content/packaging/cpbest10.html>, 2000-06-02, Abruf am 2000-08-12.
- [AnKe2000b] *Anderson, T., McKell, M. (Hrsg.)*: IMS Content Packaging Information Model, Version 1.1. <http://www.imsproject.org/content/packaging/cpinfo11.html>, 2000-12-08, Abruf am 2001-01-04.
- [AnKe2000c] *Anderson, T., McKell, M. (Hrsg.)*: IMS Content Packaging XML Binding, Version 1.1. <http://www.imsproject.org/content/packaging/cpbind11.html>, 2000-12-08, Abruf am 2001-01-04.
- [AnKe2000d] *Anderson, T., McKell, M. (Hrsg.)*: IMS Content Packaging Best Practice Guide, Version 1.1. <http://www.imsproject.org/content/packaging/cpbest11.html>, 2000-12-08, Abruf am 2001-01-04.
- [AnKe2001] *Anderson, T., McKell, M. (Hrsg.)*: IMS Reusable Competency Definitions Information Model, Version 1.01. <http://www.imsproject.org/rcd/rcdinfo01.html>, 2001-03, Abruf am 2001-06-04.
- [AnWa2000a] *Anderson, T., Wason, T.*: IMS Learning Resource Meta-data Information Model, Version 1.1. <http://www.imsproject.org/metadata/mdinfov1p1.html>, 2000-06-05, Abruf am 2000-08-12.
- [AnWa2000b] *Anderson, T., Wason, T.*: IMS Learning Resource XML Binding Specification, Version 1.1. <http://www.imsproject.org/metadata/mdbindv1p1.html>, 2000-06-05, Abruf am 2000-08-12.
- [AnWa2000c] *Anderson, T., Wason, T.*: IMS Meta-Data Best Practice and Implementation Guide, Version 1.1. <http://www.imsproject.org/metadata/mdbestv1p1.html>, 2000-06-05, Abruf am 2000-08-12.
- [ARIA2000a] *ARIADNE*: ARIADNE Project Description. <http://ariadne.unil.ch/project/>, Abruf am 2000-05-22.
- [ARIA2000b] *ARIADNE*: ARIADNE Tools Description. <http://ariadne.unil.ch/tools/>, Abruf am 2000-05-22.
- [Arib2000] *Ariba Inc.*: cXML User's Guide Version 1.1. <http://www.cxml.org/home/index.htm>, 2000, Abruf am 2000-10-26.
- [ASTD2001] *American Society for Training & Development: E-Learning Certification Standards*. http://workflow.ecc-astdinstitute.org/index.cfm?sc=help&screen_name=cert_view, Abruf am 2001-07-31.
- [Bako1998] *Bakos, Y.*: Towards Friction-Free Markets: The Emerging Role of Electronic Marketplaces on the Internet. In: Communications of the ACM 41 (1998).
- [Balz1996] *Balzert, H.*: Lehrbuch der Software-Technik: Software-Entwicklung. In: Spektrum, Akademischer Verlag, Heidelberg, Berlin 1996.
- [Balz1998] *Balzert, H.*: Lehrbuch der Software-Technik: Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung. In: Spektrum, Akademischer Verlag, Heidelberg, Berlin 1998.
- [BAOL2001] *British Association for Open Learning*: Overview of the BAOL Quality Mark for Learning Centres. <http://www.baol.co.uk/qmlcover.doc>, Abruf am 2001-11-01.
- [BaPa1994] *Baumgartner, P., Payr, S.*: Lernen mit Software, Reihe Digitales Lernen. Österreichischer StudienVerlag, Innsbruck 1994.
- [Bate2001] *Bates, P.J.*: Digital TV und Video. In: [AdCP2001].
- [Baum 1997] *Baumgartner, P.*: Didaktische Anforderungen an (multimediale) Lernsoftware. In: [IsK11997].
- [Baum1998] *Baumgartner, P.*: Lehr- und Lernqualität von Internetanwendungen. Klagenfurt 1998. <http://www.uni-klu.ac.at/~pbaumgar/deutsch/publikationen/frpublikationen.htm>, Abruf 1999

- [BBBD1996] *Boles, D., Becker, A., Bley, S., Dauelsberg, M., Eßer, A., Knoblich, C., Kölling, M., Logemann, D., Mertins, G., Prusch, T., Steen, B., Unbehaun, S., Vosskamp, R.*: Zwischenbericht der Projektgruppe Multimedia-Präsentationen im Gesundheitswesen. Carl von Ossietzky Universität Oldenburg, Fachbereich Informatik, April 1996.
- [BBL2000] *Bodendorf, F., Bauer, C., Langenbach, C., Schertler, M., Uelpenich, S.*: Vorlesung auf Abruf im Internet. In: PIK Praxis der Informationsverarbeitung und Kommunikation, München 2000.
- [Beni1956] *Benington, H.D.*: Production of Large Computer Programs. In: Proc. ONR Symp. Advanced Programming Methods for Digital Computers. 1956.
- [Berl2001] *Berlecon Research GmbH*: Wachstumsmarkt E-Learning: Anforderungen, Akteure und Perspektiven im deutschen Markt, August 2001. <http://www.berlecon.de>.
- [BeSP1999] *Best, C., Shiels, P., de Paola, M.*: Global Education Multimedia Server – GEM. Proc. of ED-MEDIA 1999, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 1999.
- [BeUh2001] *Berger, S., Uhlenbrock, C.*: Goal Based Scenarios – ein Überblick. <http://www.wipaed.wiso.uni-goettingen.de/~jsiemon/gbs.html>, Abruf am 2001-08-23.
- [BiKö2001] *Bick, M.H., Körner, F.*: Towards Collaborative Electronic Education Markets. Projektbericht, Wirtschaftsinformatik der Universität Essen 2001.
- [Bloo1956] *Bloom, B.S.*: Taxonomy of educational objectives, Handbook. In: *Mc Kay, D. (Hrsg.): Cognitive Domain*. David Mc Kay, New York 1956.
- [Bloo1973] *Bloom, B.S.*: Taxonomie von Lernzielen im kognitiven Bereich. 3. Aufl., Beltz, Weinheim 1973.
- [Blum1998] *Blumstengel, A.*: Entwicklung hypermedialer Lernsysteme. Dissertation, Universität Paderborn, 1998. http://dsor.uni-paderborn.de/organisation/blum_diss/main_index_tour.html, Abruf am 1999-09-02.
- [BMec2001] *BMEcat*: Spezifikation BMEcat, Version 1.2. <http://www.bmecat.org/Download/BMEcatV12.pdf>, Abruf am 2001-09-07.
- [Bode1990] *Bodendorf, F.*: Computer in der fachlichen und universitären Ausbildung. In: *Endres, A., Krallmann, H., Schnupp, P. (Hrsg.): Handbuch der Informatik 15*. Oldenbourg Verlag, München, Wien 1990.
- [Boec1974] *Boeckmann, K. (Hrsg.): Lernziele und Erfolgskontrolle*. 10. Aufl., Klinkhardt, Bad Heilbrunn/Obb. 1974.
- [Boeh1986] *Boehm, B.*: A Spiral Model of Software Development and Enhancement. In: ACM SIGSOFT 11 (1986).
- [BoJR1998] *Booch, G., Jacobson, I., Rumbaugh, J.*: The Unified Modeling Language Reference Manual. Addison-Wesley, New York 1998.
- [Borl2001] *Borland Software Corporation*: Borland JBuilder 5, Online Documentation. <http://www.borland.com/techpubs/jbuilder/jbuilder5/index1024x768-ent.html>, Abruf am 2001-04-02.
- [BoSc1995] *Borghoff, U., Schlichter, J.*: Rechnergestützte Gruppenarbeit. Springer, Berlin, Heidelberg, New York 1995.
- [BrDr1995] *Bröhl, A.-P., Dröschel, W.*: Das V-Modell. 2. Aufl, Oldenbourg Verlag, München 1995.
- [Brem1998] *Bremer, G.*: Genealogie von Entwicklungsschemata. In: [KnMO1998].

- [Brem2000] *Bremer, C.*: Virtuelles Lernen in Gruppen: Rollenspiele und Online-Diskussionen und die Bedeutung von Lerntypen. In: *Scheuermann, F. (Hrsg.): Campus 2000 – Lernen in neuen Organisationsformen*. Münster 2000.
- [Brow1985] *Brown, J.S.*: Process versus Product: A Perspective on Tools for Communal and Informal Electronic Learning. In: *Journal of Educational Computing Research* 2 (1985) 1.
- [Brun1961] *Bruner, J.S.*: The Act of Discovery. In: *Harvard Educational Review* 31 (1961).
- [Brun1966] *Bruner, J.S.*: Towards a theory of instruction. Newton, New York 1966.
- [Brus2000] *Brusilovsky, P.*: Concept-Based Courseware Engineering for Large Scale Web-based Education. In: *Bordeau, J., Heller, R. (Hrsg.): Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications*. AACE, Charlottesville, VA 2000.
- [Burg1997] *Burger, C.*: Groupware: Kooperationsunterstützung für verteilte Anwendungen. 1.Aufl., dpunkt – Verlag für digitale Technologie, Heidelberg 1997.
- [Bußm1998] *Bußmann, N.*: Erfolgsrezept Sozialkompetenz. In: *Manager Seminare* 8 (1998) 32.
- [CALI1998] *CALIBER-NET-Consortium*: A guide for developers of Open & Distance Learning. Projektbericht, W.J.K. Davies & Guildford Educational Services 1998.
- [Caro1990] *Carroll, J. M.*: The Nurnberg Funnel: Designing minimalist instruction for practical computer skills. MIT Press, Cambridge, MA 1990.
- [CeAC1996] *Cennamo, K.S., Abell, S.K., Chung, M.*: A "layers of negotiation" model for designing constructivist learning materials. In: *Educational Technology* 36(1996) 4.
- [CEHE2001] *Consortium for Excellence in Higher Education*: Mapping the QAA Framework and the Excellence Model, FINAL PROJECT REPORT GMP 143/QAA, Sheffield Hallam University, July 2001.
- [CENI2000a] *CEN/ISS Learning Technologies Workshop*: A Standardization Work Programme for "Learning and Training Technologies & Educational Multimedia Software" CWA, Version 4.5. <http://www.cenorm.be/iss/Workshop/lt/draft-final-report/cwa4-5.pdf>, 2000-07-14, Abruf am 2000-07-20.
- [CENI2000b] *CEN/ISS Workshop Metadata for Multimedia Information – Dublin Core*: CWA 13988, DELIVERABLE D2, GUIDANCE INFORMATION FOR THE USE OF DUBLIN CORE IN EUROPE. CEN, Brüssel 2000.
- [Chro1992] *Chroust, G.*: Modelle der Software-Entwicklung. 1. Aufl., Oldenbourg Verlag, München 1992.
- [Clan1993] *Clancey, W.*: Situated Action: A neuropsychological interpretation, Response to Vera and Simon. In: *Cognitive Science* 17 (1993).
- [Clar1994] *Clarke, R.E.*: Media will never influence learning. *Educational Technology Research and Development*, 42 (1994) 2.
- [Clar1999] *Clark, R.*: Electronic Commerce Definitions. <http://www.anu.edu.au/people/Roger.Clark/EC/ECDefns.html>, 1999, Abruf am 2000-09-15.
- [Clea2001] *o.V.*: CLEAR Campus Übersicht. <http://www.iaws.sowi.uni-bamberg.de/forschung/projekte/lernsoft/clear-campus/index.html>, Abruf am 2001-09-01.

- [Codo2001] *Codone, S.*: Measuring Quality in the Production of Web-based Training: Instrucional Design, Process Control, and User Satisfaction. Interservice/Industry Training, Simulation and Education Conference, Orlando, FL, Nov 2001.
- [CoEa1995] *Cousins, J.B., Earl, L.*: Participatory Evaluation in Education. Falmer Press, London 1995.
- [Conk1987] *Conklin, J.*: Hypertext: An Introduction and Survey. In: IEEE Computer. 20 (1987) 9.
- [Cort1974] *de Corte, E.*: Analyse der Lernzielproblematik. In: [Boec1974].
- [CoVe1999a] *Collier, G., Veres W.*: IMS Enterprise Information Model, Version 1.01. <http://www.imsproject.org/enterprise/eninfo03.html>, 1999-12-21, Abruf am 2000-08-12.
- [CoVe1999b] *Collier, G., Veres W.*: IMS Enterprise XML Binding Specification, Version 1.01. <http://www.imsproject.org/enterprise/enbind03.html>, 1999-12-21, Abruf am 2000-08-12.
- [Craw2001] *Crawley, R.M.*: A Comparison of Computer-Supported Collective Learning Application. <http://www.bton.ac.uk/cscl/jtap/paper2.htm>. Abruf am 2001-04-11.
- [CTGV1990] *Cognition & Technology Group at Vanderbilt*: Anchored instruction and its relationship to situated cognition. In: Educational Researcher, 19 (1990) 6.
- [CTGV1993] *Cognition & Technology Group at Vanderbilt*: Anchored instruction and situated cognition revisited. In: Educational Technology, 33 (1993) 3.
- [Cupl2000] *Currie, D., Place, C.*: Learning Object Containers: a suggested method of transporting metadata with a learning object. In: *Bordeau, J., Heller, R. (Hrsg.)*: Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2000.
- [CXML 2001] *Commerce XML: Commerce XML Users Guide*. <http://www.cxml.org/current/cXMLUsersGuide.pdf>, Abruf am 2001-05-04.
- [Dali1993] *Dalin, P.*: How Schools Improve. Cassell, London 1993.
- [Dave1968] *Dave, R.*: Eine Taxonomie pädagogischer Ziele und ihre Beziehung zur Leistungsmessung. In [InMa1968].
- [DCMI1999] *Dublin Core Metadata Initiative*: Dublin Core Metadata Element Set Reference Description, Version 1.1. http://purl.org/dc/documents/proposed_recommendations/pr-dces-19990702.htm, 1999-07-02, Abruf am 1999-12-10.
- [DDVC2001] *Van Durm, R., Duval, E., Verhoeven, B., Cardinaels, K., Olivié, H.*: Extending The ARIADNE Web-Based Learning Environment. In: *Montgomerie, C., Viteli, J. (Hrsg.)*: Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2001.
- [Dede1998] *Dedering, H.*: Pädagogik in der Arbeitswelt. 1. Aufl., Beltz, Weinhiem 1998.
- [DeDe1999] *Deitel, H. M. Deitel, P.J.*: JAVA How to program, 3. Aufl., Prentice Hall, Upper Saddle River, NJ 1999.
- [DEKL2001] *Dippe, G., Eltén, A., Kollia, V., Lindholm, J., Lindström, B., Tsakarissianos, G.*: Research on Quality Assessment Management and Selection Criteria regarding Content for Schools. Project Report, Version 1.0, European Treasury Browser, IST-1999-11781, D3.1, WP3, 2001.
- [Diek1995] *Diekmann, A.*: Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen. 1. Aufl.,rororo, Hamburg 1995.

- [Diep1999] *Diepold, P.*: Lernen und Lehren in der Informationsgesellschaft WS 1999/2000. <http://www.educat.hu-berlin.de/~diepold/vorlesung/index.html>, Abruf am 1999-10-13
- [Diet1998] *Dietrich, T.*: Zeit- und Grundfragen der Pädagogik: eine Einführung in pädagogisches Denken. 8. Aufl., Klinkhardt, Bad Heilbrunn/Obb. 1998.
- [Dodd2001a] *Dodds, P.(Hrsg.)*: Advanced Distributed Learning Initiative - Sharable Courseware Object Reference Model – SCORM, Version 1.1. http://www.adlnet.org/scorm/docs/SCORM_2.doc, 2000-01-16, Abruf am 2001-07-25.
- [Dodd2001b] *Dodds, P.(Hrsg.)*: Advanced Distributed Learning Initiative - Sharable Content Object Reference Model – The SCORM Overview, Version 1.2. http://www.adlnet.org/library/documents/scorm/specifications/SCORM_1.2_Overview.pdf, 2001-10-01, Abruf am 2001-11-01.
- [Dodd2001c] *Dodds, P.(Hrsg.)*: Advanced Distributed Learning Initiative - Sharable Content Object Reference Model – The SCORM Content Aggregation Model, Version 1.2. http://www.adlnet.org/library/documents/scorm/specifications/SCORM_1.2_CAM.pdf, 2001-10-01, Abruf am 2001-11-01.
- [Dodd2001d] *Dodds, P.(Hrsg.)*: Advanced Distributed Learning Initiative - Sharable Content Object Reference Model – The SCORM Run-Time Environment, Version 1.2. http://www.adlnet.org/library/documents/scorm/specifications/SCORM_1.2_RunTimeEnv.pdf, 2001-10-01, Abruf am 2001-11-01.
- [Doms1980] *Domsch, M.*: Systemgestützte Personalarbeit. 1. Aufl., Gabler, Wiesbaden 1980.
- [Dows1987] *Dowson, M.*: Iteration in the Software Process. Review of the 3rd Int. Software Process Workshop. In: Proceedings of the 9th Int. Conf. On Software Engineering. IEEE Computer Society Press, Washington 1987.
- [Dres1997] *Drescher, R.*: Didaktik: Sinnmitte unterrichtlichen Handelns; Grundlegung, Unterrichtsprinzipien, Handlungsempfehlungen. 1. Aufl., Wifa, Ansbach 1997.
- [Drum1995] *Drumm, H.J.*: Personalwirtschaftslehre. 3. Aufl., Springer, Berlin 1995.
- [DSST1997] *Dijkstra, S., Seel, N.M., Schott, F., Tennyson, D.*: Instructional Design, International Perspectives. Volume 2 Solving instructional design problems. Lawrence Erlbaum, Hillsdale 1997.
- [DuLJ1993] *Duffy, T.M., Lowyck, J., Jonassen, D.H. (Hrsg.)*: Designing Environments for Constructive Learning. NATO ASI Series. Series F: Computer and Systems Sciences, 1205, Springer, Berlin, Heidelberg 1993.
- [Duva2001] *Duval, E.*: Standardized Metadata for Education: a Status Report. In: *Montgomerie, C., Viteli, J. (Hrsg.)*: Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2001.
- [EBXM2000] *Electronic Business XML*: Catalog of Core Processes for Electronic Business XML (ebXML). http://www.ebxml.org/project_teams/business_process/wip/ccbp-analysis/COREPROCESS_ebxml.doc, Abruf am 2000-11-29.
- [ECML1999] *ECML Alliance*: Electronic Modeling Language specifications. <http://www.ecml.org/specifications.html>, 1999, Abruf am 2000-10-25.
- [EDUC1999] *EDUCAUSE*: IMS Learning Resource Meta-data Best Practises and Implementation Guide, Version 1.0. <http://www.imsproject.org/metadata/mdbest01.html>, Abruf am 1999-10-12.
- [Eiff1991] *von Eiff, W.*: Prozesse optimieren – Nutzen erschließen. In: IBM Nachrichten 41 (1991).
- [Eisn1985] *Eisner, E.*: The Art of Educational Evaluation. Falmer Press, Lewes 1985.

- [EuCo2000] *European Commission: Indicators and Benchmarks of Quality of School Education, European Report on Quality of School Education – Sixteen Quality Indicators.* <http://europa.eu.int/comm/education/indic/backen.html>, 2000, Abruf am 2001-01-23.
- [Eule1992] *Euler, D.: Didaktik des Computerunterstützten Lernens.* In: *Holz, H., Zimmer, G. (Hrsg.): Multimediales Lernen in der Berufsbildung 3.* BW Verlag, Nürnberg 1992.
- [Eule1998] *Euler, D.: Kommunikationsfähigkeit und Computerunterstütztes Lernen.* In: *Twardy, M. (Hrsg.): Wirtschafts-, Berufs- und Sozialpädagogische Texte 13.* Müller-Botermann Verlag, Köln 1989.
- [Fels1996] *Felsch, A.: Personalentwicklung und organisationales Lernen: mikropolitische Perspektiven zur theoretischen Grundlegung.* 1. Aufl., S + W Steuer- und Wirtschaftsverlag, Hamburg 1996.
- [FEMS1997] *Friedrich, H.F., Eigler, G., Mandl, H., Schnotz, W., Schott, F., Seel, N.M. (Hrsg.): Multimediale Lernumgebungen in der betrieblichen Weiterbildung: Gestaltung, Lernstrategien und Qualitätssicherung.* Luchterhand, Neuwied, Kriftel, Berlin 1997.
- [FeSc2001] *Ferstl, O.K., Schmitz, K.: Integrierte Lernumgebungen für virtuelle Hochschulen.* *Wirtschaftsinformatik 43 (2001) 1.*
- [FHSU2000] *Ferstl, O.K., Hahn, K., Schmitz, K., Ullrich, C.: Funktionen und Architektur einer Internet-basierten Lernumgebung für individuelles und kooperatives Lernen.* In: *Uellner, S., Wulf, V. (Hrsg.): Vernetztes Lernen mit digitalen Medien.* Physica-Verlag, Heidelberg 2000.
- [FiBM1998] *Fischer, T., Biskup, H., Müller-Luschnat, G.: Begriffliche Grundlagen für Vorgehensmodelle.* In: [KnMO1998].
- [Fick1992] *Fickert, T.: Multimediales Lernen: Grundlagen, Konzepte, Technologien.* Deutscher Universitätsverlag, Wiesbaden 1992.
- [Fitz1996] *Fitz-Gibbon, C.T.: Monitoring Education: Indicators, Quality and Effectiveness.* Cassell, London 1996.
- [FIBS1978] *Flehsig, K.-H., Burfeind, H., Schmidt, W.: Erstfassung eines Katalogs didaktischer Modelle.* Göttinger Monographien zur Unterrichtsforschung, Zentrum für didaktische Studien e.V., Angerstein 1978.
- [Flec1996] *Flehsig, K.-H.: Kleines Handbuch didaktischer Modelle.* 4. Aufl., Neuland – Verlag für lebendiges Lernen, Eichenzell 1996.
- [FIHa1973] *Flehsig, K.-H., Haller, H.D.: Entscheidungsprozesse in der Curriculumentwicklung, Band 24 der Reihe Gutachten und Studien der Bildungskommission.* 1. Aufl., Klett, Stuttgart 1973.
- [FIHa1977] *Flehsig, K.-H., Haller, H.D.: Einführung in didaktisches Handeln. Ein Lehrbuch für Einzel- und Gruppenarbeit.* 2. Aufl., Klett, Stuttgart 1977.
- [Fras1981] *Fraser, R.: Design and evaluation of educational software for group presentation.* In: *Howe, J.A.M., Ross, R.M. (Hrsg.): Microcomputers in Secondary Education.* Kogan Page, London 1981.
- [FrBa1997] *Friedrich, H.F., Ballstaedt, S.-P.: Strategien für das Lernen mit Medien.* In: [FEMSSS1997].
- [Frey1971] *Frey, K. (Hrsg.): Kriterien in der Curriculumskonstruktion.* 2. Aufl., Beltz, Weinheim 1971.
- [Frey1971a] *Frey, K.: Kriteriensysteme in der Curriculumskonstruktion: begriffliche Grundlagen.* In: [Frey1971].
- [Frey1972] *Frey, K.: Theorien des Curriculums.* 2. Aufl., Beltz, Weinheim 1972.

- [Fric1997] *Fricke, R.*: Evaluation von Multimedia. In: [IsK11997].
- [Full1992] *Fullan, M.*: Successful School Improvement. Open University Press, Buckingham 1992.
- [GaBe1996] *Gage, N.L., Berliner, D.C.*: Pädagogische Psychologie (übersetzt aus dem englischen von G. Bach, Originaltitel: Educational Psychology). Psychologie Verlags Union, Weinheim 1996.
- [GaBW1979] *Gagne, R.M., Briggs, L.J., Wagner, R.*: Principles of Instructional Design. 2. Aufl., Holt, Rinehart, & Winston, New York 1979.
- [Gaed2000] *Gaede, B.*: Die Instructional Material Description Language und das Produktionssystem für interaktive Lernsoftware ProfiL. Eine Metaspezifikationsprache und ein Generator für Lernsoftware. 10. Arbeitstreffen der GI-Fachgruppe Intelligente Lehr-/Lernsysteme, Hamburg 2000.
- [Gagn1965] *Gagne, R.M.*: The Conditions of Learning and Theory of Instruction. 1. Aufl., Holt, Rinehart, & Winston, New York 1965.
- [Gagn1980] *Gagne, R.M.*: Die Bedingungen des menschlichen Lernens. 5. Aufl., Schroedel Verlag, Hannover 1980.
- [GaPM1993] *Garzotto, F., Paolini, P., Mainetti, L.*: Navigation Patterns in Hypermedia Databases. In: Proceedings of the 26th Hawaii International Conference on System Science. IEEE Computer Society Press 1993.
- [GaPS1993] *Garzotto, F., Paolini, P., Schwabe, D.*: HDM: A model-based approach to hypertext application design. In: ACM Transactions on Information Systems, 11(1993) 1.
- [Gaug92] *Gaugler, E. (Hrsg.)*: Handwörterbuch des Personalwesens. 2. Aufl., Schäffer-Poeschel, Stuttgart 1992.
- [GeIn2001] *Gesellschaft für Informatik e.V.*: Informatik-Begriffsnetz. <http://www.vorgehensmodelle.de/Giak/>, Abruf am 04-13-2001.
- [Geiß1996] *Geißler, H. (Hrsg.)*: Arbeit, Lernen und Organisation. 1. Aufl., Deutscher Studien Verlag, Weinheim 1996.
- [GeMa1994] *Gerstenmaier, J., Mandl, H.*: Wissenserwerb unter konstruktivistischer Perspektive. Forschungsbericht Nr. 33, Ludwig-Maximilians-Universität, München 1994.
- [Geut1989] *Geuting, M.*: Planspiele und soziale Simulation im Bildungsbereich. In: *Pöggeler, F. (Hrsg.)*: Studien zur Pädagogik, Andragogik und Gerontagogik, Band 10. Peter Lang, Frankfurt a. M. 1989.
- [GhVo1997] *Ghazanfari, N., Voshmgir, S.*: Verfahrensvergleich ARIS vs. SOM. <http://www.wu-wien.ac.at/usr/h92/h9250216/SOM.1/inhalt.html>, Abruf am 1999-12-11.
- [Gies1999] *Giesecke, H.*: Einführung in die Pädagogik. 5. Aufl., Juventa, Weinheim 1999.
- [GiMA1998] *GiMA*: Definition und Geschichte - Was ist ein Planspiel? <http://www.planspiel.de/planspiel/defi.htm>, Abruf am 1999-01-22.
- [GIKu2000] *Glötz, P., Kubicek, H.*: Finanzierung virtueller Studienangebote. In: Bertelsmann Stiftung, Heinz-Nixdorf-Stiftung: Studium online, Verlag Bertelsmann Stiftung, Gütersloh 2000.
- [GISc1992a] *Glowalla, U., Schoop, E. (Hrsg.)*: Hypertext und Multimedia. Neue Wege in der computerunterstützten Aus- und Weiterbildung. Springer Verlag, Heidelberg 1992.

- [GIsc1992b] *Glowalla, U., Schoop, E.*: Entwicklung und Evaluation computerunterstützter Lehrsysteme. In: [GIsc1992a].
- [GoPr1999] *Goldfarb, C. F., Prescod, P.*: XML-Handbuch. Prentice Hall, München 1999.
- [Gray1998] *Grayling, T.*: and loathing of the help menu: A usability test of online help. In: Technical Communication 45 (1998) 2.
- [Gree1989] *Greeno, J.G.*: Situations, mental models, and generative knowledge. In: *Klahr, D., Kotovsky, K.* (Hrsg.): Complex information processing: The impact of H.A. Simon. Erlbaum, Hillsdale, NJ 1989.
- [Gree1992] *Greeno, J.G.*: The situation in cognitive theory: Some methodological implications of situativity. In: Proc. of Meeting of APS. San Diego 1992.
- [Grob1997] *Grob, H.L.*: Ereignisorientierte Planspiele. <http://www-wi.uni-muenster.de/aw/plans/planspiel>, Abruf am 1999-01-22.
- [GrSe1996] *Grob, H.L., Seufert, S.*: Vorgehensmodelle bei der Entwicklung von CAL-Software. <http://www.wi.uni-muenster.de/aw/calcat/ab5/>, Abruf am 1999-11-15.
- [GrWa1997] *Griffin, S., Wason, T.*: The year of metadata. In: Educom Review 32 (1997) 6.
- [Guy2001] *Guy, R.*: Jext – Overview. <http://www.jext.org/index.html>, Abruf am 2001-11-01.
- [Häfe2001] *Häfele, H.*: eLearning Plattformen. <http://www.virtual-learning.at/community.htm>, Abruf am 2001-09-04.
- [Hämä1997] *Hämäläinen, M.*: Course Brokers for Customised On-Demand Training. In: Proc. Of ENABLE 1997, Espoo, Finland 1997.
- [Hämä1999] *Hämäläinen, M.*: Enabling Electronic Markets for Education and Training. In: Proc. of ENABLE 1999, Espoo, Finland. <http://www.enable.evitech.fi/Enable99/papers/hamalainen/hamalainen.html>, Abruf am 2000-03-02.
- [Hami1976] *Hamilton, D.*: Curriculum Evaluation. Open Books, London 1976.
- [Harg1994] *Hargreaves, A.*: Changing Teachers, Changing Times: Teachers' Work and Culture in the Postmodern Age. Cassell, London 1994.
- [Harl1994] *Harlen, W.*: Enhancing Quality in Assessment. Paul Chapman, London 1994.
- [Hase1995] *Hasebrook, J.P.*: Lernen mit Multimedia. In: Pädagogische Psychologie 9 (1995) 2.
- [HaSt1990] *Hawes, H., Stephens, D.*: Questions of Quality. Logman, London 1990.
- [HaSt1993] *Hartmann, H., Sternberg, R.J.*: A Broad BACEIS for Improving Thinking. In: Instructional Science 5 (1993) 21.
- [HäWV1996] *Hämäläinen, M., Whinston, A.B., Vishik, S.*: Electronic Markets for Learning: Education Brokerages on the Internet. In: Communications of the ACM 39 (1996) 2.
- [Hayn1996] *Haynes, M.E.*: Projektmanagement: Von der Idee bis zur Umsetzung; der Projekt-Lebenszyklus, Faktor Qualität/Zeit/Kosten, erfolgreicher Abschluß. 1. Aufl., Ueberreuter Wirtschaftsverlag, Wien 1996.
- [HeEd1990] *Henderson-Seller, B., Edwards, J.M.*: Object oriented software systems life cycle. In: CACM 33 (1990) 9.

- [Heer1996] *Heeren, E.*: Technology Support for Collaborative Distance Learning. CTIT Ph.D.-thesis series, Enschede 1996.
- [HEFC2001] *Higher Education Funding Council for England, Quality Assurance Agency for Higher Education*: Quality assurance in higher education - Proposals for consultation. http://www.hefce.ac.uk/Pubs/hefce/2001/01_45/01_45.doc, Abruf am 2001-11-01.
- [HeHä2000] *Heinrich, L.J., Häntschel, I.*: Evaluation und Evaluationsforschung in der Wirtschaftsinformatik. Handbuch für Praxis, Lehre und Forschung, Oldenbourg Verlag, München, Wien 2000.
- [Hent94] *Hentze, J.*: Personalwirtschaftslehre I: Grundlagen, Personalbedarfsermittlung, -beschaffung, -entwicklung und -einsatz. 6. Aufl., UTB, Bern, Stuttgart, Wien 1994.
- [Hein2000] *Heinrich, L.J.*: Bedeutung von Evaluation und Evaluationsforschung in der Wirtschaftsinformatik. In: [HeHä2000].
- [HeMe1994] *Heeg, F.-J., Meyer-Dohm, P. (Hrsg.)*: Methoden der Organisationsgestaltung und Personalentwicklung: Vorgehensweisen, Methoden und Techniken bei der Umsetzung von Lean-Management-Konzepten und der Einführung gruppenorientierter Strukturen. 1. Aufl., Hanser, München 1994.
- [HePI2000] *Hepburn, G., Place, C.*: Learning Objects: Communicating the Pedagogical Potential. In: *Bordeau, J., Heller, R. (Hrsg.)*: Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2000.
- [HeRo1989] *Heinrich, L.J., Roithmayr, F.*: Wirtschaftsinformatik-Lexikon. 3. Aufl., Oldenbourg Verlag, München, Wien 1989.
- [HeSO1965] *Heimann, P., Schulz, W., Otto, G.*: Unterricht - Analyse und Planung. Schroedel, Hannover 1965.
- [Hess1998] *Hesse, W.*: Vorgehensmodelle für objektorientierte Software-Entwicklung. In: [KnMO1998].
- [HeSW2001] *Henze, N., Schmidt, C., Wolpers, M.*: Mediengestützte Didaktik für qualitative Methoden in der Sozialforschung auf der Basis semantischer Modellierung. Virtueller Campus, 19. - 21.9.2001, Hildesheim, Germany.
- [Hirs1974] *Hirschmann, G.*: Lernzielorientierte Unterrichtsvorbereitung in der Weiterbildung. In: [Boec74], S. 124-137.
- [HKLR1984] *Hesse, W., Heutgen, H., Luft, A.L., Rombach, H.D.*: Ein Begriffssystem für die Softwaretechnik. In: Informatik-Spektrum 7 (1984).
- [HNWW2001] *Hadhami, D., Nejd, W., Wolf, B., Wolpers, M.*: Open Learning Repositories and Metadata Modeling., <http://www.semanticweb.org/SWWS/program/full/paper38.pdf>, Abruf am 2001-11-22.
- [HoCo1999] *Horstmann, C., Cornell, G.*: Core Java 2 - Band 1. Markt + Technik 1999.
- [Hoff1997] *Hoffmann, K.*: Anchored Instruction. <http://pc06-paedglo.psychol.uni-giessen.de/osinet/paedagog/instrukt/konstruk/anchinst.htm>, 1997, Abruf am 1999-01-22.
- [HoKn1996] *Hoffmann, J., Knopf, M.*: Der Erwerb formaler Schlüsselqualifikationen. In: [Wein1996].
- [Holl1986] *Holly, P.*: Developing a professional evaluation. In: Cambridge Journal of Education 16 (1986) 2.
- [Hopk1989] *Hopkins, D.*: Evaluation for School Development. Open University Press, Milton Keynes 1989.

- [Horv1994] *Horváth, P.*: Controlling. 5. Aufl., Vahlen, München 1995.
- [Hump1989] *Humphrey, W.S.*: Managing the Software Process. Addison-Wesley, Reading 1989.
- [HüRS1999] *Hümpel, C., Renner, T., Schmitz, V.*: Spezifikation BMEcat, Version 1.0. <http://www.bmecat.org>, Abruf am 1999-11-08.
- [HüSc2001] *Hümpel, C., Schmitz, V.*: BMEcat - Produktkataloge austauschen per XML. In: *Turowski, K., Fellner, K. (Hrsg.): XML in der betrieblichen Praxis: Standards, Möglichkeiten, Praxisbeispiele*, dpunkt-Verlag, Heidelberg 2001.
- [InMa1968] *Ingenkamp, K., Marsolek, T. (Hrsg.):* Möglichkeiten und Grenzen der Testanwendung in der Schule. 1. Aufl., Beltz, Weinheim 1968.
- [IsK11997] *Issing, L., Klimsa, P. (Hrsg.):* Information und Lernen mit Multimedia 2. Beltz Psychologie-Verlags-Union, Weinheim, Basel 1997.
- [IsSB1995] *Isakowitz, T., Stohr, E.A., Balasubramanian, P.*: RMM: A Methodologie for Structured Hypermedia Design. In: *Communications of the ACM* 38(1995) 8.
- [Issi1997] *Issing, L.*: Instruktionsdesign für Multimedia. In: [IsK11997].
- [JaMe1994] *Jank, W., Meyer, H.*: Didaktische Modelle. 3. Aufl., Cornelsen, Berlin 1994.
- [JoGr1993] *Jonassen, D.H., Grabowski, B.L.*: Handbook of Individual Differences, Learning, and Instruction. Lawrence Erlbaum, Hillsdale NJ 1993.
- [JoMM1993] *Jonassen, D.H., Mayes, T., McAleese, R.*: A manifesto for a constructive approach to uses of technology in higher education. In: *Duffy, T.M., Lowyck, J., Jonassen, D.H., Welsh, T.M. (Hrsg.): Designing environments for constructive learning*. Springer, Berlin 1993.
- [KaSa1999] *Kaplan, S., Sawhney, M.*: B2B E-Commerce Hubs: Towards a Taxonomy of Business Models. <http://gsbwww.uchicago.edu/fac/steven.kaplan/research/taxonomy.pdf>, 1999, Abruf am 2000-02-02.
- [KDKS2001] *Konstantopoulos, M., Darzentas, J.S., Koutsabasis, P., Spyrou, T., Darzentas, J.*: Towards Integration of Learning Objects Metadata and Learner Profiles Design: Lessons Learnt from GESTALT. In: *Journal on Interactive Learning Environments, Special issue on educational metadata*, 2001.
- [Kell1987] *Keller, J.M.*: The Systematic Process of Motivational Design. In: *Performance & Instruction* 11/12 (1987).
- [Kerr2000] *Kerres, M.*: Multimediale und telemediale Lernumgebungen - Konzeption und Entwicklung. 2. Aufl., Oldenbourg Verlag, München, Wien 2000.
- [KhBu1995] *Khoshafian, S., Buckiewicz, M.*: Introduction to Groupware, Workflow and Workgroup Computing. 1. Aufl., John Wiley & Sons, New York 1995.
- [Kien1994] *Kienbaum, J. (Hrsg.):* Visionäres Personalmanagement. 2. Aufl., Schäffer-Poeschel, Stuttgart 1994.
- [KiGn1999] *Kien, M., Gneiting, S.*: Mobilfunk der dritten Generation. In: *Funkschau* 70 (1999) 7.
- [KiPa1997] *Kinshuk, Patel, A.*: A Conceptual Framework for Internet based Intelligent Tutoring Systems. In: *Knowledge Transfer* 2 (1997).
- [KiPR2001] *Kinshuk, Patel, A., Russell, D.*: Intelligent and Adaptive Systems. In: [AdCP2001].

- [Klei1998] *Klein, A.*: Teleteaching scenarios for high bandwidth networks. In: Computer Networks and ISDN systems 30 (1998).
- [KlSt2001] *Klein, M., Stucky, W.*: Ein Vorgehensmodell zur Erstellung virtueller Bildungsinhalte. In: Wirtschaftsinformatik 43 (2001).
- [KnMO1998] *Kneuper, R., Müller-Luschnat, G., Oberweis, A.*: Vorgehensmodelle für die betriebliche Anwendungsentwicklung. Teubner, Stuttgart 1998.
- [KoHB2000] *Koppi, T., Hodgson, L., Bayly, J.*: The Often Missing but Essential Component for Online Learning: a Learning Resource Catalogue. In: *Bordeau, J., Heller, R. (Hrsg.): Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications.* AACE, Charlottesville, VA 2000.
- [KoHo2001] *Koppi, T., Hogson, L.*: Universitas 21 Learning Resource Catalogue using IMS Metadata and a New Classification of Learning Objects. In: *Montgomerie, C., Viteli, J. (Hrsg.): Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications.* AACE, Charlottesville, VA 2001.
- [Kolb1981] *Kolb, D.A.*: Learning Styles and Disciplinary Differences. In: Chickering, A.W. and Associates: The Modern American College. Jossey-Bass Publishers, San Francisco, Washington, London 1981.
- [Köll2001] *Kölling, M.*: Teaching Java with BlueJ – A Sequence of Assignments. <ftp://mars.pscit.monash.edu.au/pub/mik/papers/bluej-assignments.pdf>, 2001, Abruf am 2001-09-04.
- [Kope2001] *Koper, R.*: Modeling units of study from a pedagogical perspective – the pedagogical meta-model behind EML. <http://eml.ou.nl/introduction/articles.htm>, 2001-06, Abruf am 2001-07-01.
- [KöRo1999] *Kölling, M., Rosenberg, J.*: Testing Object-Oriented Programs: Making it Simple. <ftp://mars.pscit.monash.edu.au/pub/mik/papers/testing.pdf>, 1999, Abruf am 2001-09-04.
- [KöRo2000] *Kölling, M., Rosenberg, J.*: BlueJ – The Hitch Hiker’s Guide to Object Orientation. <ftp://mars.pscit.monash.edu.au/pub/mik/papers/hitch-hiker.pdf>, 2000, Abruf am 2001-09-04.
- [KöRo2001] *Kölling, M., Rosenberg, J.*: Guidelines for Teaching Object Orientation with Java. <ftp://mars.pscit.monash.edu.au/pub/mik/papers/java-guidelines.pdf>, 2001, Abruf am 2001-09-04.
- [Kräm1999] *Kraemer, W.*: Education Brokerage - Wissensallianzen zwischen Hochschulen und Unternehmen. Information Management & Consulting 14 (1999).
- [KrBM1964] *Krathwohl, D.R., Bloom, B.S., Masia, B. B.*: Taxonomy of educational objectives, Handbook II: Affective Domain. David McKay, New York 1964.
- [Krem1997] *Kremer, A.*: Didaktik. In: [Bern97].
- [Krop1997] *Kropp, W.*: Systematische Personalwirtschaft: Wege zu vernetzt – kooperativen Problemlösungen. 1. Aufl., Oldenbourg Verlag, München 1997.
- [KrWe1995] *Kraus, G., Westermann, R.*: Projektmanagement mit System - Organisation, Methoden, Steuerung. 1. Aufl., Gabler, Wiesbaden 1995.
- [KrBM1975] *Krathwohl, D.R., Bloom, B.S., Masia, B.B.*: Taxonomie von Lernzielen im affektiven Bereich. 1. Aufl., Beltz, Weinheim 1975.
- [KSHM2001] *Kleinberger, T., Schrepfer, L., Holzinger, A., Müller, P.*: A Multimedia Repository for Online Educational Content. In: *Montgomerie, C., Viteli, J. (Hrsg.): Proc. of EDMEDIA 2001, World*

- Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2001.
- [Kuge2001] *Kugemann, W.*: Fragebogen zur Qualitätsprüfung, Projektbericht, FIM Psychologie, Universität Erlangen-Nürnberg 2001.
- [LaBo1998] *Langenbach, C., Bodendorf, F.*: An Education Broker Toolset for Web-Course Customization. In: *Journal of Universal Computer Science* 4 (1998) 10.
- [LaBo2000] *Langenbach, C., Bodendorf, F.*: Electronic Education Mall – A Virtual Service Center for Distance Learning. In: *International Journal of Electronic Commerce* 4 (2000).
- [LaMM2001] *Larsen, K., Morris, R., Martin, J.P.*: Trade in educational services: Trends and Emerging Issues. <http://ishi.lib.berkeley.edu/cshe/projects/university/ebusiness/KLarsen.html>, Abruf am 2001-09-01.
- [Land01] *Landon, B.*: Comparative analysis of online educational delivery applications. <http://www.c2t2.ca/landonline/>, Abruf am 2001-08-30.
- [LDKG1974] *Levin, J.R., Divine-Hawkins, P., Kerst, S.M., Guttman, J.*: Individual differences in learning from pictures and words: The development and application of an instrument. In: *Journal of Educational Psychology* 66(1974), S. 296-303.
- [LiMe1994a] *Li, Z., Merrill, M.D.*: An Instructional Design Expert System. In: [MeTw1994].
- [LiMe1994b] *Li, Z., Merrill, M.D.*: ID Expert 2.0: Design Theory and Process. In: [MeTw1994].
- [Lind2001a] *Lindner, R.*: Specification of an Open Experiencing Platform Core for IT-supported Learning, Education, and Training, PROMETEUS SIG-DESIGN, First Draft, Version 01 July 2001. <http://www.gris.informatik.tu-darmstadt.de/~lindner/PROMETEUS/SIG-DESIGN/Discussion/Spec-FD-2001-07-01.doc>, Abruf am 2001-07-03.
- [Lind2001b] *Lindner, R.*: Design of Electronic Learning Environments. http://www.gris.informatik.tu-darmstadt.de/~lindner/MoU/ELE_Design.html, Abruf am 2001-09-01.
- [Litk1995] *Litke, H.D.*: Projektmanagement: Methoden, Techniken, Verhaltensweisen. 3. Aufl., Hanser, München 1995.
- [LTSC2000a] *IEEE Learning Technology Standards Committee*: Draft Standard for Computer-Managed Instruction (CMI). <http://ltsc.ieee.org/doc/wg11/CMI-Sem34.doc>, 2000-09, Abruf am 2001-05-30.
- [LTSC2000b] *IEEE Learning Technology Standards Committee*: Draft Standard for Learning Technology – Public and Private Information (PAPI) for Learners (PAPI Learner), Version 7.0. http://edutool.com/papi/papi_learner_07_main.doc, 2000-11-28, Abruf am 2001-05-30.
- [LTSC2001a] *IEEE Learning Technology Standards Committee*: P1484.1/D8, Draft Standard for Learning Technology – Learning Technology Systems Architecture, Version 8.0. <http://ltsc.ieee.org/wg1>, 2001-05-06, Abruf am 2001-07-01.
- [LTSC2001b] *IEEE Learning Technology Standards Committee*: IEEE P1484.3/D3 Draft Standard for Information Technology - Learning Technology – Glossary. <http://ltsc.ieee.org/doc/wg3/Glossary-20010309.doc>, Abruf am 2001-07-03.
- [LTSC2001c] *IEEE Learning Technology Standards Committee*: Draft Standard for Learning Object Metadata, Version 6.1. <http://ltsc.ieee.org/doc/wg12/LOMv6.1.html>, 2001-05-12, Abruf am 2001-05-30.

- [LTSC2001d] *IEEE Learning Technology Standards Committee*: IEEE P1484.3/D3, Draft Standard for Information Technology - Learning Technology – Glossary, <http://ltsc.ieee.org/doc/wg3/Glossary-20010309.doc>, Abruf am 2001-06-09.
- [Lust2001] *Lusti, M.*: Data Warehousing and Data Mining. Eine Einführung in Entscheidungsunterstützende Systeme. 2. Aufl., Springer, Hamburg 2001.
- [Mada1994] *Madauss, B.*: Handbuch Projektmanagement: mit Handlungsanleitungen für Industriebetriebe, Unternehmensberater und Behörden. 5. Aufl., Schäffer-Poeschel, Stuttgart 1994.
- [Mage62] *Mager, R. F.*: Lernziele und programmierter Unterricht. 2. Aufl., Beltz, Weinheim 1962.
- [MaGR1997] *Mandl, H., Gruber, H., Renkl, A.*: Situiertes Lernen in multimedialen Lernumgebungen. In: [IsK11997].
- [MaVa1987] *Maturana, H.R., Varela, F.J.*: Der Baum der Erkenntnis. Die biologischen Wurzeln des menschlichen Erkennens. Scherz Verlag, Bern, München 1987.
- [MCJa1996] *McCormick, R., James, M.*: Curriculum Evaluation in Schools. 2. Aufl. Routledge, London 1996.
- [Mell2000] *Mellis, W.*: Lehre als Evaluationsobjekt. Einführung und Grundlegung. In: [HeHä2000].
- [Merr1983] *Merrill, M.D.*: Component Display Theory. In: *Reigeluth, C.M. (Hrsg.)*: Instructional Design Theories and Models: An Overview of their Current Status. Laurence Erlbaum, Hillsdale 1983.
- [Merr1987] *Merrill, M.D.*: An expert system for instructional design. In: IEEE Expert 2 (1987).
- [Merr1993] *Merrill, M.D.*: An integrated model for automating instructional design and delivery. In: *Spector, M., Polson, M., Muraida, D.J. (Hrsg.)*: Automating instructional design: Concepts and issues. Educational Technology Publications, Englewood Cliffs 1993.
- [Merr1994] *Merrill, M.D.*: The Descriptive Component Display Theory. In: [MeTw1994].
- [MeTw1994] *Merrill, M.D., Twitchell, D.G. (Hrsg.)*: Instructional Design Theory. Educational Technology Publications, Englewood Cliffs 1994.
- [Metz1973] *Metzger, P.W.*: Managing a programming project. Prentice Hall, Upper Saddle River, NJ 1973.
- [Meye1974] *Meyer, H.L.*: Einführung in die Curriculum-Methodologie. 2. Aufl., Kösel, München 1974.
- [Meye1975] *Meyer, H. L.*: Trainingsprogramm zur Lernzielanalyse. 2. Aufl., Fischer, Frankfurt a.M. 1975.
- [Meye1996] *Meyer, B.*: Erfolgsschlüssel Objekttechnologie. Hanser, München 1996.
- [Mich1999] *Michel, T.*: XML kompakt: Eine praktische Einführung. Hanser, München 1999.
- [MiSI1996] *Mizoguchi, R., Sinitsa, K., Ikeda, M.*: Task ontology design for intelligent educational/training systems. In: Proc. of Workshop "Architectures and Methods for designing Cost-Effective and Reusable ITSs". Third International Conference on Intelligent Tutoring Systems, ITS-96, Montreal, June 12-14, 1996.
- [Mizo1993] *Mizoguchi, R.*: Knowledge Acquisition and Ontology. Proceedings of KB&KS'93, 1993.
- [Möll1973] *Möller, C.*: Technik der Lernplanung: Methoden und Probleme der Lernzieleerstellung. 4. Aufl., Beltz, Weinheim 1973.

- [Moon1999] Moonen, J.: The design and prototyping of digital learning materisl: Some new perspectives. In: Van den Akker, J., Branch, R., Gustafson, K., Nieveen, N.M., Plomp, T. (Hrsg.): Design approaches and tools in education and training. Kluwer, Dordrecht 1999.
- [Moon2000] Moonen, J.: A three-space design strategy for digital learning material. In: Educational Technology 40 (2000) 2.
- [Moon2001] Moonen, J.: Design Methodology. In: [AdCP2001].
- [MoSt1990] Mortimore, P., Stone, C.: Measuring Educational Quality. In: British Journal of Educational Studies 39 (1990) 1.
- [MüSe1994] Müller, W., Seibt, D.: Methoden- und computergestützte Personalplanung. 2. Aufl., Datakontext, Köln 1994.
- [Murr1999] Murray, T.: Authoring Intelligent Tutoring Systems: An analysis of the state of the art. In: International Journal of Artificial Intelligence in Education 10 (1999).
- [Netq2000] Netquest: Tutorial Markup Language (TML). http://www.ilrt.bris.ac.uk/netquest/liveserver/TML_INSTALL/doc/tml.html, Abruf am 2000-08-25.
- [Neu1991] Neuberger, O.: Personalentwicklung. 1. Aufl., Enke Stuttgart 1991.
- [Neul1999] Neuland, M.: Neuland-Moderation. 3. Aufl., Neuland – Verlag für lebendiges Lernen, Künzell 1999.
- [Nils1980] Nilsson, J.: Principles of Artificial Intelligence. Berlin, Springer 1980.
- [Noki2001] o. V.: Nokia and INSEAD together with ICUS and StarHub launch the world's first module of WAP enabled E-Learning. http://www.icus.net/pages/researchnews/nokia_insead.htm, Abruf am 2001-02-19.
- [NoSc1999] Noack, W., Schienmann, B.: Objektorientierte Vorgehensmodelle im Vergleich. In: Informatik-Spektrum 22 (1999).
- [OECD1989] OECD: Schools and Quality: An International Report. OECD, Paris 1989.
- [OECD1995] OECD: Performance Standards in Education: In Search of Quality. OECD, Paris 1995.
- [Oest1999] Oesterreich, B.: Objektorientierte Softwareentwicklung: Analyse und Design mit der Unified modeling language. 4. Aufl. – 1. korrigierter Nachdruck, Oldenbourg Verlag, München 1999.
- [OLea2000] O'Leary, D.E.: Supply Chain Processes and Relationships for Electronic Commerce. In: Shaw, M., Blanning, R., Strader, T., Whinston, A. (Hrsg.): Handbook on Electronic Commerce, Springer, Berlin 2000.
- [OISt1998] Olfert, K., Steinbuch, P.A.: Personalwirtschaft. 7. Aufl., Kiehl, Ludwigshafen 1998.
- [Oppe2001] Oppermann, R.: User Interface Design. In: [AdCP2001].
- [OTPC1998] Open Trading Protocol Consortium: Internet Open Trading Protocol Part 2: Specifications, V 0.9. <http://www.otp.org>, 1998, Abruf am 2000-10-25.
- [PaAC1999] Paquette, G., Aubin, C., Crevier, F.: MISA, A Knowledge-based Method for the Engineering of Learning Systems, In: Journal of Courseware Engineering, 2 (1999).

- [PaAd2001a] *Pawlowski, J.M., Adelsberger, H.H.*: Standardisierung von Lerntechnologien. In: *Wirtschaftsinformatik* 43 (2001).
- [PaAd2001b] *Pawlowski, J.M., Adelsberger, H.H.*: Electronic Business and Education. In: [AdCP2001].
- [PaBV2001] *Pawlowski, J.M., Bick, M.H., Veith, P.*: An Architecture of an Electronic Education Market Using XML-Standards. In: *Montgomerie, C., Viteli, J. (Hrsg.): Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2001.*
- [Paiv1971] *Paivio, A.*: Imagery and verbal processes. Holt, Rinehart & Winston, New York 1971.
- [Pani2001] *Panitz, T.*: Collaborative Versus Cooperative Learning-A Comparison Of The Two Concepts Which Will Help Us Understand The Underlying Nature Of Interactive Learning. <http://www.capecod.net/~tpanitz/tedspage/tedsarticles/coopdefinition.htm>, Abruf am 2001-04-11.
- [Paqu2001] *Paquette, G.*: Designing Virtual Learning Centers. In: [AdCP2001].
- [PaRa1997] *Patzak, G., Rattay, G.*: Projektmanagement: Leitfaden zum Management von Projekten, Projektportfolios und projektorientierten Unternehmen. 2. Aufl., Linde, Wien 1997.
- [Paul1995] *Paulsen, M.F.*: The Online Report on Pedagogical Techniques for Computer-Mediated Communication. <http://www.hs.nki.no/~morten/cmcped.htm>, 1995, Abruf 1999-10-10.
- [Pawl2000] *Pawlowski, J.M.*: The Essen Learning Model – A Multi-Level Development Model. In: *Staff and Educational Development International* 4 (2000).
- [Pawl2001] *Pawlowski, J.M.*: The Essen Learning Model – Homepage. <http://elm.wi-inf.uni-essen.de>, Abruf am 2001-11-01.
- [PfSt1997] *Pfohl, H.C., Stölzle, W.*: *Planung und Kontrolle: Konzeption, Gestaltung, Implementierung*. 2. Aufl., Vahlen, München 1997.
- [PhMe2000] *Phillips, C., Meeker, M.*: The B2B Internet Report. <http://www.msdc.com/techresearch/b2b/info.html>, 2000, Abruf am 2000-08-09.
- [Piag1954] *Piaget, J.*: The construction of reality in the child. Ballentine, New York 1954.
- [Piag1985] *Piaget, J.*: The equilibration of cognitive structure. University of Chicago Press, Chicago 1985.
- [Pico1982] *Picot, A.*: Transaktionskostenansatz in der Organisationstheorie. In: *DBW* 42 (1982) 2.
- [Piet1998] *Pietsch, T., Martiny, L., Klotz, M.*: Strategisches Informationsmanagement: Bedeutung und organisatorische Umsetzung. 3. Aufl., Erich Schmidt, Berlin 1998.
- [Plan2001] o.V.: Planspiele - Definition und Geschichte. <http://www.planspiel.de/planspiel/defi.htm>, Abruf am 2001-06-23.
- [Popl2001] *Poplawski, D.A.*: Objects Have Class: An Introduction to Programming. McGraw-Hill, New York 2001.
- [PoWi2000] *Pott, O., Wielage, G.*: XML Praxis und Referenz. 2. Aufl., Markt&Technik, München 2000.
- [Prei1995] *Preiß, P.*: Göttinger Katalog didaktischer Modelle von K.-H. Flechsig. <http://www.wiso.gwdg.de/~ppreiss/didaktik/method96.html>, 1995, Abruf am 1998-11-12.

- [PRKO1998] *Patel, A., Russell, D., Kinshuk, Oppermann, R., Rashev, R.*: An initial framework of contexts for designing usable intelligent tutoring systems. In: *Information Services and Use* 18 (1998).
- [QAAH1999] *Quality Assurance Assocoation for Higher Education*: Guidelines on the Quality Assurance of Distance Learning. <http://www.qaa.ac.uk/dlg>, 1999, Abruf am 2000-08-01.
- [Raas1993] *Raasch, J.*: Systementwicklung mit strukturierten Methoden: ein Leitfaden für Praxis und Studium. 3. Aufl., Hanser, München, Wien 1993.
- [Rahm1999] *Rahm, E.*: Informationsmodellierung mit dem Entity-Relationship-Modell. <http://www.informatik.uni-leipzig.de/ifi/abteilungen/db/skripte/DBS1/HTML/kap3-1.html>, Abruf am 1999-12-11.
- [Rapp2000] *Rappa, M.*: Business Models on the Web. http://ecommerce.ncsu.edu/business_models.html, 2000, Abruf am 2000-09-08.
- [RaSv1995] *Rayport, J.F., Sviokla, J.J.*: Managing in the Marketspace. In: *Harvard Business Review* 72 (1995) 6.
- [Räth2001] *Räth, C.*: UMTS macht mobiles Lernen möglich. In: *ntz - Informationstechnik + Telekommunikation* 54 (2001) 3.
- [ReBM1994] *Reigeluth, C.M., Bunderson, C.V., Merrill, M.D.*: Is There a Design Science of Instruction. In: [MeTw1994].
- [Redd2001] *Redding, G.A.*: From DL POTS to DL PANS: The Advance Distributed Learning (ADL) Initiative. In: *Montgomerie, C., Viteli, J. (Hrsg.): Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications*. AACE, Charlottesville, VA 2001.
- [Rekk1998] *Rekkedal, T.*: Quality Assessment and Evaluation - Basic Philosophies, Concepts and Practices at NKI, Norway, http://www.fernuni-hagen.de/ZIFF/ziff108/rek_rep2.doc, Abruf am 2000-19-09.
- [ReMa1994] *Reinmann-Roth, G., Mandl, H.*: Wissensvermittlung: Ansätze zur Förderung des Wissenserwerbs. Forschungsbericht Nr. 34, Ludwig-Maximilians-Universität, München 1994.
- [ReRa2001] *Reeves, T.C., Raven, A.*: Performance-Support Systems. In: [AdCP2001].
- [ReTe2000] *Requisite Technology*: Electronic Catalog XML (ecx) Specification. <http://www.ecx-xml.org>, 2000, Abruf am 2000-10-25.
- [RHEW2001] *Rueß, C., Hördt, J., Eberhardt, R., Wolf, M., Wilk, C.*: Architectural Concept for an Integrated Learning Platform Using Learning Object Metadata. In: *Montgomerie, C., Viteli, J. (Hrsg.): Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications*. AACE, Charlottesville, VA 2001.
- [Riek82] *Rieken, H.*: Nebenberufliche Qualifikation. Lerneinheit 1 Lernziele. In: *Arbeitskreis Universitäre Erwachsenenbildung in Zusammenarbeit mit der Universität Oldenburg*. Beltz, Weinheim 1982.
- [Rigg1998] *Riggert, W.*: Betriebliche Informationskozepte: von Hypertext zu Groupware. 1. Aufl., Vieweg, Braunschweig 1998.
- [Riis1995] *Riis, J.O. (Hrsg.): Simulation Games and Learning in Production Management*. Chapman & Hall, London 1995.
- [RiNu1994] *Riley, K.A., Nuttall, D.L. (Hrsg.): Measuring Quality: Education Indicators – United Kingdom and International Perspectives*. Falmer Press, London 1994.

- [Rinz1995] *Rinza, P.*: Projektmanagement: Planung, Überwachung und Steuerung von technischen und nichttechnischen Vorhaben. 3. Aufl., VDI-Verlag, Düsseldorf 1994.
- [RJM1995] *Riis, J.O., Johansen, J., Mikkelsen, H.*: Design of simulation games. In: [Riis1995].
- [Roh1967] *Robinson, S.B.*: Bildungsreform als Revision des Curriculums. 1. Aufl., Luchterhand, Neuwied 1967.
- [Roh2000] *Robson, R.*: Report on Learning Technology Standards, In: *Bordeau, J., Heller, R. (Hrsg.): Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2000.*
- [Roh2001] *Rohrbacher, K., Götz, A.*: IMT-2000 - Der Traum vom 3G-Weltstandard. In: *Funkschau 72 (2001) 3.*
- [Roh92] *Rohn, W.E.*: Simulation – Praxis am Modell erlernen. In: *Graf, J. (Hrsg.): Planspiele – simulierte Realitäten für den Chef von morgen. Gabal, Speyer 1992, S. 19-28.*
- [Ros1994] *Rossi, G., Schwabe, D.*: From Domain Models to Hypermedia Applications. Workshop of Methodologies for Designing and Developing Hypermedia Applications, Edinburgh, September 1994.
- [Ros1995] *Rossi, G., Schwabe, D.*: Building Hypermedia Applications as Navigational Views of Information Models. In: *Proceedings of Annual Hawaii International Conference on System Science 28 (1995).*
- [Ros1995] *von Rosenstiel, L.*: Organisationspsychologie. 8. Aufl., Kohlhammer 1995.
- [Roy1970] *Royce, W.W.*: Managing the development of large software systems. *Proceedings of IEEE Wescon 1970.*
- [Ruh95] *Ruohomäki, V.*: Viewpoints on learning and education with simulation games. In: [Riis1995].
- [Rus1999] *Russell, T.*: The No Significant Difference Phenomenon – A Comparative Research Annotated Bibliography on Technology for Distance Education. International Distance Education Certification Center Montgomery, AL 1999.
- [Rus2001] *Russell, T.*: The "No Significant Difference Phenomenon", <http://teleeducation.nb.ca/nosignificantdifference/>, Abruf am 2001-10-10.
- [Sab2000] *Saba Software Inc.*: Universal Learning Format (ULF), Technical Specification, Version 1.0. <http://www.saba.com/standards/ulf/Pdf/ulfSpecification.pdf>, 2000-10-23, Abruf am 2001-01-02.
- [Sas1997] *Sander, J., Scheer, A.-W.*: Offene Lernumgebungen in der Aus- und Weiterbildung am Beispiel des PPS-Trainers. In: *Scheer, A.-W. (Hrsg.): Veröffentlichungen des Institut für Wirtschaftsinformatik (IW) an der Universität des Saarlandes, Heft 140, 1997.*
- [SBH2001] *Seufert, S., Back, A., Häusler, M.*: E-Learning: Weiterbildung im Internet. SmartBooks, Kilchberg 2001.
- [Sche1995] *Scheer, A.-W., Nüttgens, M., Zimmermann, V.*: Rahmenkonzept für ein integriertes Geschäftsprozeßmanagement. In: *Wirtschaftsinformatik 3 (1995).*
- [Sche1998a] *Scheer, A.-W.*: ARIS. In: *Bernus, P., Mertins, K., Schmidt, G. (Hrsg.): Handbook on Architectures of Information Systems. Springer, Heidelberg 1998.*
- [Sche1998b] *Scheer, A.-W.*: ARIS - vom Geschäftsprozeß zum Anwendungssystem. 3. Aufl., Springer, Berlin 1998.

- [Sche1998c] *Scheer, A.-W.*: ARIS - Modellierungsmethoden, Metamodelle, Anwendungen. 3. Aufl., Springer, Berlin 1998.
- [Sche2001a] *Scheer, A.-W.*: Corporate Universities drängen auf den Bildungsmarkt. In: *GMD Projektträger Neue Medien in der Bildung (Hrsg.): Anytime, Anywhere - IT-gestütztes Lernen in den USA*. http://www.gmd.de/PT-NMB/e-learning_in_USA/Bericht.pdf, Abruf am 2001-09-01.
- [Sche2001b] *Scheer, A.-W.*: Fernuniversitäten erschließen den internationalen Bildungsmarkt. In: *GMD Projektträger Neue Medien in der Bildung (Hrsg.): Anytime, Anywhere - IT-gestütztes Lernen in den USA*. http://www.gmd.de/PT-NMB/e-learning_in_USA/Bericht.pdf, Abruf am 2001-09-01.
- [Schi1995] *Schierenbeck, H.*: Grundzüge der Betriebswirtschaftslehre. 12. Aufl., Oldenbourg Verlag, München 1995.
- [Schl1997] *Schlichter, J.*: Vorlesung ‚Computergestützte Gruppenarbeit‘. <http://www11.informatik.tu-muenchen.de/lehre/lectures/cscw-WS97/CSCW-top-WS9798.html>, Abruf am 1999-11-01.
- [Scho1999] *Scholz, C. (Hrsg.)*: Innovative Personalorganisation: Center-Modelle für Wertschöpfung, Strategie, Intelligenz und Virtualisierung. 1. Aufl., Luchterhand, Neuwied 1999.
- [Schm2000] *Schmitz, K.*: Konzeption eines Application Framework zur Virtualisierung wirtschaftswissenschaftlicher Lehr-/Lernsituationen. Dissertation. Deutscher Universitätsverlag, Wiesbaden 2000.
- [Schu1996] *Schulmeister, R.*: Grundlagen hypermedialer Lernsysteme: Theorie – Didaktik – Design. Addison-Wesley, Bonn 1996.
- [Schu2000] *Schulte, C.*: Vom Modellieren zum Gestalten –Objektorientierung im Informatikunterricht. Informatik - Ausbildung und Beruf 2000, http://mm-info.uni-paderborn.de/didaktik/Veroeffentlichungen/modellieren_gestalten.pdf, Abruf am 2001-05-01.
- [Schu2001] *Schulmeister, R.*: Virtuelle Universität. Virtuelles Lernen. Oldenbourg Verlag, München 2001.
- [Schw1998] *Schwarze, J.*: Informationsmanagement: Planung, Steuerung, Koordination und Kontrolle der Informationsversorgung im Unternehmen. 1. Aufl., Verlag Neue Wirtschafts – Briefe, Herne 1998.
- [ScZe1995] *Schaub, H., Zenke, K.G.*: Wörterbuch zur Pädagogik. 1. Aufl., dtv, München 1995.
- [SeBH2001] *Seufert, S. Back, A., Häusler, M.*: E-Learning - Weiterbildung im Internet. Das „Plato-Cookbook“ für internetbasiertes Lernen. Smartbooks Publishing, Kilchberg 2001.
- [SeDi1997] *Seel, N.M., Dijkstra, S.*: General Introduction. In: [DSST1997].
- [Seif1995] *Seifert, J.W.*: Visualisieren, Präsentieren, Moderieren. 8. Aufl., Gabal, Offenbach 1995.
- [Seif1999] *Seifert, J.W.*: Besprechungs-Moderation. 5. Aufl., Gabal, Offenbach 1999.
- [Seuf1999] *Seufert, S.*: PLATO –„Electronic Cookbook“ for Internet-based Learning. In: Proc. of HCI. München, 1999.
- [ShMO1991a] *Shavelson, R.J., McDonell, L., Oakes, J.*: What are Educational Indicators and Indicator Systems? In: Practical Assessment, Research & Evaluation 2 (1991) 11.
- [ShMO1991b] *Shavelson, R.J., McDonell, L., Oakes, J.*: Steps in Designing an Indicator System. In: Practical Assessment, Research & Evaluation 2 (1991) 12.

- [SIIA2000] *Software & Information Industry Association: Schools Interoperability Framework Implementation Specification, V1.0.* <http://www.sifinfo.org/spec.html>, 2000-07-27, Abruf am 2001-02-01.
- [Simp1966] *Simpson, J.S.:* The classification of educational objectives, psychomotor domain. Office of Education Project No. 5-85-104, University of Illinois, Urbana 1996.
- [Skin1954] *Skinner, B.F.:* The Science of Learning and the Art of Teaching. In: Harvard Educational Review 24 (1954).
- [Skin1958] *Skinner, B.F.:* Teaching Machines. In: Science 128 (1958).
- [SmSh2001a] *Smythe, C., Shepherd, E.:* IMS Question & Test Interoperability Information Model Specification, Version 1.1. <http://www.imsproject.org/question/qtinfo03.html>, 2001-03-09, Abruf am 2001-06-25.
- [SmSh2001b] *Smythe, C., Shepherd, E.:* IMS Question & Test Interoperability XML Binding Specification, Version 1.1. <http://www.imsproject.org/question/qtbind03.html>, 2001-03-09, Abruf am 2001-06-25.
- [SmSh2001c] *Smythe, C., Shepherd, E.:* IMS Question & Test Interoperability Best Practice & Implementation Guide, Version 1.1. <http://www.imsproject.org/question/qtibpig03.html>, 2001-03-09, Abruf am 2001-06-25.
- [SmTR2001] *Smythe, C., Tansey, F., Robson, R.:* IMS Learner Information Package, Best Practice & Implementation Guide, Version 1.0. <http://www.imsproject.org/profiles/lipbest01.html>. 2001-03-09. Abruf am 2001-04-01.
- [Spit1996] *Spitzer, D.R.:* Motivation: The Neglected Factor in Instructional Design. In: Educational Technology 5-6 (1996).
- [Squi1997] *Squires, D.:* An heuristic approach to the evaluation of educational multimedia software, In: Proceedings of CAL 97 Conference. University of Exeter, Exeter, UK 1997.
- [SSRF1999] *Steinacker, A., Seeberg, C., Reichenberger, K., Fischer S., Steinmetz, R.:* Dynamically Generated Tables of Contents as Guided Tours in Adaptive Hypermedia Systems. In: Proc. of EDMEDIA 1999, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 1999.
- [Stae94] *Staehele, W. H.:* Management: Eine verhaltenswissenschaftliche Perspektive. 7. Aufl., Vahlen, München 1994.
- [Step1997] *Steppi, H.:* CBT- Computer Based Training: Planung, Design und Entwicklung interaktiver Lernprogramme. Klett, Stuttgart 1997.
- [Stri2001] *Strijker, A.:* Using metadata for re-using material and providing user support tools. In: *Montgomerie, C., Viteli, J. (Hrsg.):* Proc. of EDMEDIA 2001, World Conference on Educational Multimedia, Hypermedia & Telecommunications. AACE, Charlottesville, VA 2001.
- [Stru1998] *Struck, O.:* Individuenzentrierte Personalentwicklung: Konzepte und empirische Befunde. 1. Aufl., Campus, Frankfurt/Main 1998.
- [Strz1997] *Strzebkowski, R.:* Realisierung von Interaktivität und multimedialen Präsentationstechniken. In: [IsKL1997].
- [StSh2000] *Strader, T.J., Shaw, M.J.:* Electronic Markets: Impact and Implications. In: *Shaw, M., Blanning, R., Strader, T., Whinston, A. (Hrsg.):* Handbook on Electronic Commerce. Springer, Berlin 2000.

- [Sues2000] *Suess, C.*: Document Type Definitions (DTDs). <http://daisy.fmi.uni-passau.de/pakmas/LM2L/>, Abruf am 2000-09-09.
- [SuFr2001] *Suess, C., Freitag, B.*: Learning Material Markup Language –LMML. IFIS-Report 2001/03, IFIS - Institut für Informationssysteme und Softwaretechnik. Universität Passau 2001.
- [SuKa2000] *Suhl, L., Kassarke, S.*: OR-World – Using Learning Objects in a Hypermedia Learning Environment. In: *Bordeau, J., Heller, R. (Hrsg.): Proc. of ED-MEDIA 2000, World Conference on Educational Multimedia, Hypermedia & Telecommunications.* AACE, Charlottesville, VA 2000.
- [Suth2000] *Suthers, D. D.*: Using Learning Object Metadata in a Database of Primary and Secondary School Resources. In Proceedings of International Conference on Computers in Education, Nov. 21-24, Taipei, Taiwan 2000.
- [Swert2000] *Swertz, C.*: Ausbildung zum Gebrauch didaktischer Ontologien. In: *Ohly, H.P., Rahmstorf, G., Sigel, A.*: Globalisierung und Wissensorganisation. Würzburg 2000.
- [THHL1996] *Tulodziecki, G., Hagemann, W., Herzig, B., Leufen, S., Mütze, C.*: Neue Medien in den Schulen: Projekte-Konzepte-Kompetenzen. Verlag Bertelsmann Stiftung, Gütersloh 1996.
- [Thom1992] *Thom, N.*: Personalentwicklung und Personalentwicklungsplanung. In: [Gaug1992].
- [Timm2000] *Timmers, P.*: Global and Local in Electronic Commerce. In: *Bauknecht, K., Kumar Madria, S., Pernul, G.*: Electronic Commerce and Web Technologies. Lecture Notes in Computer Science, 1875. Springer, Berlin 2000.
- [Trig2001] *Trigari, M.*: Multilingual Thesaurus, why?, European Treasury Browser, http://www.eun.org/eun.org2/eun/en/etb/content_frame.cfm?lang=en&ov=3813, Abruf am 2001-11-01.
- [Twig1997] *Twigg, C.*: Putting Learning on Track. In: *Educom Review 32 (1997) 6*, <http://www.educause.edu/pub/er/edreview.html>, Abruf am 1998-06-06.
- [URAC2000] *US Regional Accrediting Commissions (RAC)*: Distance Education Statement and Guidelines. <http://www.ncacih.org/resources/draftdistanceguide/index.html>, 2000, Abruf am 2001-07-01.
- [Voßb1989] *Vofßbein, R.*: Organisation. 3. Aufl., Oldenbourg Verlag, München 1989.
- [Voßb1990] *Vofßbein, R.*: Management der Bürokommunikation: strategische und konzeptionelle Gestaltung von Bürokommunikationssystemen. 1. Aufl., Vieweg, Braunschweig 1990.
- [Voßb1995] *Vofßbein, R. (Hrsg.)*: Organisation sicherer Informationsverarbeitungssysteme: Konzepte und Lösungen. 1. Aufl., Oldenbourg Verlag, München 1995.
- [VRee2000] *Van Reeken, A.J.*: Informationssysteme als Evaluationsobjekt. Einführung und Grundlegung. In: [HeHä2000].
- [Vygo1978] *Vygotsky, L.*: Mind in society: the development of higher psychological processes. Harvard University Press, Cambridge MA, 1978.
- [Wats1930] *Watson, J.B.*: Behaviorism. Norton, New York 1930.
- [WeCl1999] *Westland, C.J., Clark, T.H.K.*: Global Electronic Commerce: Theory and Case Studies. MIT Press, Cambridge, MA 1999.
- [Weid1993] *Weidenmann, B.*: Wissenserwerb mit Bildern. In: Gelbe Reihe: Arbeiten zur Empirischen Pädagogik und Pädagogischen Psychologie 27 (1993).

- [Wein1996a] *Weinert, F.E.(Hrsg.):* Psychologie des Lernens und der Instruktion. In: *Birnbaumer, N., Frey, D., Kuhl, J., Prinz, W., Weinert, F.E.:* Enzyklopädie der Psychologie, Band 2. Hogrefe, Göttingen 1996.
- [Wein1996b] *Weinert, F.E.:* Lerntheorien und Instruktionsmodelle. In: [Wein1996a].
- [Will2000] *Willis, J.:* The maturing of constructivist instructional design: Some basis principles that can guide practice. *Educational Technology* 40(1996) 1, S. 5-16.
- [Wint2001] *Winter, M.-A.:* TU Berlin: Mobiles Internet auf dem Campus. <http://www.teltarif.de/arch/2001/kw17/s5017.html>, Abruf am 2001-09-14.
- [WeAl1992] *Welge, M., Al-Laham, A.:* Planung: Prozesse – Strategien - Maßnahmen. 1. Aufl., Gabler, Wiesbaden 1992.
- [WiSc1990a] *Wiedemeyer, G.R., Schuster, M.:* Risiken moderner Informationstechnologie für die Beteiligungsrechte der Arbeitnehmer. In: [Bell1990].
- [WiSc1990b] *Wiedemeyer, G.R., Schuster, M.:* Die Anwendung des Bundesdatenschutzgesetzes auf das Arbeitsverhältnis und die Frage eines bereichsspezifischen Personaldatengesetzes. In [Bell90]2.
- [Wiss2000] *Wissenschaftszentrum Berlin:* Homepage des Wissenschaftszentrums Berlin, <http://www.wz-berlin.de>, Abruf am 2000-01-21.
- [Wöhe1996] *Wöhe, G.:* Einführung in die allgemeine Betriebswirtschaftslehre. 19. Aufl., Vahlen, München 1996.
- [WWWC1999a] *World Wide Web Consortium:* HTML 4.01 Specification. <http://www.w3.org/TR/1999/REC-html401-19991224>, 1999, Abruf am 2001-02-01.
- [WWWC1999b] *World Wide Web Consortium:* Resource Description Framework (RDF) Model and Syntax Specification. <http://www.w3.org/TR/REC-rdf-syntax/>, 1999-02-22, Abruf am 2000-06-06.
- [WWWC1999c] *World Wide Web Consortium.* XSL Transformations (XSLT), Version 1.0. <http://www.w3.org/TR/1999/REC-xslt-19991116.html>, 1999-11-16, Abruf am 2000-01-11.
- [WWWC2000] *World Wide Web Consortium.* Extensible Markup Language (XML) 1.0 (Second Edition). <http://www.w3.org/TR/2000/WD-xml-2e-2000-08-14>, 2000-08-14, Abruf am 2000-08-25.
- [WWWC2001a] *World Wide Web Consortium:* XML Linking Language (XLink), Version 1.0. <http://www.w3.org/TR/2000/REC-xlink-20010627/>, 2001-06-27. Abruf am 2001-08-08.
- [WWWC2001b] *World Wide Web Consortium.* Extensible Stylesheet Language (XSL), Version 1.0. <http://www.w3.org/TR/2001/REC-xsl-20011015/>, 2001-10-15, Abruf am 2001-11-01.
- [Xala2001] *o. V.:* Xalan-C++ version 1.2. <http://xml.apache.org/xalan-c/index.html>, Abruf am 2001-09-24.
- [Ziec1974] *Ziechmann, J.:* Der technologische Ansatz zur Lernzielfindung. In: [Boec1974].

Lebenslauf

PERSÖNLICHE ANGABEN

| | |
|---------------------|----------------------|
| Name | Jan Martin Pawlowski |
| Familienstand | verheiratet |
| Staatsangehörigkeit | deutsch |
| Geburtstag | 28.04.1971 |
| Geburtsort | Essen |

AUSBILDUNG

| | |
|-----------|---|
| 1981-1990 | Helmholtz-Gymnasium Essen, Abschluss: Allgemeine Hochschulreife |
| 1990-1995 | Studium Wirtschaftsinformatik DI an der Universität Essen Abschluss: Diplom-Wirtschaftsinformatiker (DI), Note: gut |
| 1995 | Diplomarbeit: „Evaluation der Faktorenoptimierung von adaptiv kombinierten Heuristiken zur Lösung von Schedulingproblemen“ |
| 1995-1996 | Studium Wirtschaftsinformatik (DII) an der Universität Essen Abschluss: Diplom-Wirtschaftsinformatiker (DII), Note: sehr gut |
| 1996 | Diplomarbeit: „Empirische Analyse des Simulated Annealing zur Lösung des Traveling Salesman Problems“ |

BERUFSTÄTIGKEIT

| | |
|-----------|--|
| 1992-1993 | Anwendungsprogrammierer Oregon Steel Mills, Portland, OR, USA |
| 1996-2001 | Wissenschaftlicher Mitarbeiter im Fachgebiet „Wirtschaftsinformatik der Produktionsunternehmen“ der Universität Essen Dissertationsthema: „Das Essener-Lern-Modell (ELM): Ein Vorgehensmodell zur Entwicklung computerunterstützter Lernumgebungen“ |
| seit 2001 | Wissenschaftlicher Mitarbeiter im Projekt „Virtuelle Aus- und Weiterbildung Wirtschaftsinformatik“ |

Eidesstattliche Erklärung

Ich versichere an Eides statt durch meine Unterschrift, dass ich die vorstehende Arbeit selbstständig und ohne fremde Hilfe angefertigt und alle Stellen, die ich wörtlich oder annähernd wörtlich aus Veröffentlichungen übernommen habe, als solche kenntlich gemacht habe, mich auch keiner anderen als der angegebenen Literatur oder sonstiger Hilfsmittel bedient habe.

Die Arbeit hat in dieser oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.

Essen, den 29.11.2001

(Jan M. Pawlowski)