

Didaktik der Systemadministration

Hubert Feyrer <hubert@feyrer.de>

28. Januar 2005

Zusammenfassung

Die Lehre der Systemadministration wird meist nur von technischer Seite betrachtet. Dieser Artikel betrachtet das Thema aus didaktischer Sicht. Der erste Teil des Artikels betrachtet psychologische Hintergründe der Didaktik und stellt verschiedene bekannte Lerntheorien sowie deren didaktische Umsetzung dar, wobei speziell auf Aspekte wie die Einbettung der Lernsysteme in verschiedene Unterrichtsformen eingegangen wird. Basierend auf diesen Grundlagen wird die Form eines idealen Unterrichts definiert. Der zweite Teil des Artikels stellt die an der Fachhochschule Regensburg existierende Vorlesung "Systemadministration" vor, und beschreibt ihr Zielpublikum, den Inhalt und das Layout der Vorlesung sowie die verwendeten didaktischen Instrumente. Die Analyse der Vorlesung unter didaktischen Aspekten zeigt bestehende Defizite und definiert die nötigen Schritte, um diese zu beheben.

Inhaltsverzeichnis

1	Bitte fragen Sie Ihren Administrator: Einleitung	2
2	Didaktische Hintergründe	2
2.1	Psychologie und Lerntheorie	2
2.2	Didaktische Umsetzung	4
2.3	Aspekte der Umsetzung	5
2.4	Idealer Unterricht	6
3	Fallstudie	7
3.1	Zielpublikum	7
3.2	Inhalt	7
3.3	Layout	10
3.4	Didaktische Instrumente	12
4	Analyse der aktuellen Situation & Ausblick	13
	Literatur	14

1 Bitte fragen Sie Ihren Administrator: Einleitung

Vor ein paar Wochen stand im elterlichen Haushalt ein neuer PC an, natürlich mit aktuellem Redmonder Betriebssystem. Nach der Einrichtung des Systems und etwas Einarbeitungszeit hatte mein Vater nur zwei Probleme zu vermelden: der EMail-Setup brauchte eine kleine Änderung, und es tauchte ein ihm bis dahin unbekannter Begriff auf – er möge sich an seinen “Administrator” wenden!

Die Frage machte deutlich, daß auch selbst “Heimanwender”-Betriebssysteme heute nicht mehr ohne fachkundiger Unterstützung durch einen “Administrator” benutzt werden können, doch woher erhält dieser sein Wissen? Der Nürnberger Trichter wurde diesbezüglich noch nicht erfunden, viele Hobby- und auch Berufs-Administratoren wurden gefordert, sich das Wissen um die Zusammenhänge im System selbst anzueignen, und nur die wenigsten durften eine fundierte Ausbildung in diesem Bereich erfahren. Die Vermutung, daß in Zukunft vermehrt Bedarf an diesen wunderbaren Helfern bestehen wird, legt nahe, sich mit der Ausbildung eben dieser näher auseinanderzusetzen.

An der Fachhochschule Regensburg existiert seit mehreren Jahren die Vorlesung “Systemadministration”, die verschiedene Gebiete dieses weitläufigen Themas vermittelt. Dieser Artikel diskutiert zum Einstieg didaktische Grundlagen, die der Lehre zugrunde liegen, und beleuchtet anschließend Struktur und Inhalt der Vorlesung “Systemadministration”, um die aktuelle Situation zu analysieren und Zukunftsperspektiven aufzuzeigen.

Diese Arbeit entstand im Rahmen der Dissertation “Sysadmin Education in the Virtual Unix-Lab” am Lehrstuhl für Informationswissenschaft der Universität Regensburg und des Virtuellen Unix Labors am Fachbereich Informatik/Mathematik der Fachhochschule Regensburg.

2 Didaktische Hintergründe

Zunächst sollen die didaktischen Hintergründe beleuchtet werden, vor denen dann der gegenwärtige Stand der Ausbildung exemplarisch betrachtet wird. Ausgehend von den Grundlagen, die Psychologie und Pädagogik in Form von Lerntheorien liefern, wird weiter auf deren didaktische Umsetzung allgemein eingegangen, sowie ihre Aspekte bei der Umsetzung und Integration in ggf. bestehende Veranstaltungen. Die Vorstellung eines (theoretisch) idealen Unterrichts bietet die Ausgangsbasis für die praktische Betrachtungen des zweiten Teils.

2.1 Psychologie und Lerntheorie

Naturwissenschaften und der naturwissenschaftliche Teil der Psychologie haben früh den Zusammenhang zwischen den Lernprozessen des menschlichen Gehirns und ihrer Wissenschaft aufgegriffen. Es entstanden daraus Theorien des menschlichen Lernens (Lerntheorie) und der Aufbereitung von Instruktionen für effizientes Lernen (Instruktionstheorie) sowie für deren Umsetzung (Instruktionsdesign). Im Folgenden sollen einige wichtige Lerntheorien und ihre Eigenschaften und Einsatzgebiete kurz erklärt werden, die nächsten Kapitel gehen näher auf die Umsetzung ein.

Mit dem von John Watson geprägten Begriff **Behaviourismus** geht die älteste bekannte Lerntheorie auf Ivan Pavlov zurück, der seinem Hund beibrachte, daß das Läuten einer Glocke Teil der Fütterung ist, mit dem Effekt, daß der Hund später auch bereits beim Läuten der Glocke mit erhöhtem Speichelfluß reagierte. Dieses einfache Verfahren wurde von Skinner und Thorndike mit Experimenten an Tauben und Katzen weiter ausgebaut, so daß die Verhaltensmuster der Tiere regelrecht "programmiert" wurden.

Zusammen mit den Ansätzen der Kybernetik, den Menschen als Maschine zu sehen und deren Funktionsweise zu verstehen, entstand die "Kybernetische Pädagogik" in der Hoffnung, den gesamten menschlichen Lernprozeß in ähnlicher Weise programmieren zu können¹.

Die Idee des behaviouristischen Unterrichts besteht in der Vermittlung kleiner Informationseinheiten, gefolgt von einer kurzen Pause um dem Schüler Zeit zum Nachdenken zu geben und anschließendem Abfragen des gelernten Stoffes. Die Antwort des Schülers wird wiederum durch eine Rückmeldung des Lehrers bestätigt, wobei verschiedene Arten der Rückmeldung (positiv, negativ) zu unterschiedlichen Ergebnissen führen². In der Praxis wird der behaviouristische Lernansatz hauptsächlich für die Vermittlung einfacher, wenig umfangreicher Stoffgebiete eingesetzt. Umfangreichere Inhalte müssen in kleine Lernschritte aufgegliedert werden, die dann üblicherweise aufeinanderfolgend vermittelt werden.

Einer der Kritikpunkte am Behaviourismus ist, daß er wenig auf die individuelle Natur des Lerners eingeht. Der **Kognitivismus** tut dies, indem der Lerner als eigenständiges Individuum verstanden wird, der vom Lehrer vermittelte Nachrichten (im Sinne von Shannon & Weaver³) empfängt und verarbeitet, wobei die Nachrichten in verschiedenen Medien transportiert werden können. Lernen wird dabei als kreativer Problemlöse-Vorgang betrachtet, bei dem der Lerner sich dem Stoffgebiet durch Anpassung und Übernehmen von Verhaltensweisen anderer aneignet. Es wird angenommen, daß das Lernresultat nicht nur durch die äußeren Einflüsse des Lehrers, sondern auch durch den internen "Zustand" (Wissen) des Schülers bestimmt wird. Das Wissen des Lernenden ist quasi die Summe all seines Verstehens und all seiner Eindrücke, v.a. auch der durch seine Umwelt⁴.

Der Ansatz des kognitivistischen Lernens eignet sich für komplexe Themen die über einfaches Faktenlernen hinausgehen. Je nach beabsichtigtem Lernerfolg können nicht nur Fakten, sondern auch Regeln und Konzepte vermittelt werden. Ungeachtet der Vorteile des Kognitivismus bleibt jedoch der generelle Aufbau der Interaktion zwischen Schüler und Lehrer wie beim Behaviourismus erhalten.

Der **Konstruktivismus** geht einen Schritt weiter. Ausgehend von der Grundidee des Kognitivismus wird der Lehrer als bestimmender externer Einfluß komplett abgeschafft, und angenommen, daß der Schüler sich frei innerhalb einer Lernumgebung bewegen kann, und durch die gesammelten Eindrücke und das Feedback auf seine Aktionen neues Wissen erlernen kann. Durch die Annahme, daß alle Lerner individuell verschieden sind, existiert im Konstruktivismus auch kein "bester" Unterricht. Um das Lernen anzuregen werden in bestimmten Umgebungen und Situationen Aufgaben gestellt, die mit Hilfe der zur Verfügung stehenden Mittel beliebig gelöst werden können und sollen. Dieser Ansatz der "Situiertheit" verhindert auch das Entstehen von sogenann-

¹[Seidel and Lipsmeier, 1989] S. 32

²[Kerres, 1998] S. 46

³[Shannon and Weaver, 1949] S. 31

⁴[Tulodziecki, 2000] S. 58

ten tragem Wissen, da Prozeduren und Fakten immer im Kontext gelernt werden⁵.

Trotz der vielen Vorteile, die der Konstruktivismus damit scheinbar bietet wird sich bei der Umsetzung im nächsten Kapitel zeigen, daß leider nicht alles so einfach ist, so daß sich in der momentan gängigen Praxis diverse **Mischformen** etabliert haben, die sowohl Komponenten aus Kognitivismus als auch Konstruktivismus vereinen. Nennenswerte Beispiele sind hier Merrills Instruktionsdesign der 2. Generation und Mandls Idee des "situierten Lernens". Ersteres ist dem erforschenden Lernen des Konstruktivismus sehr nahe, wobei eine verstärkte Lerner-Führung durch untereinander kommunizierende, sich dem Lernerverhalten anpassende Einzelkomponenten erzeugt wird, die je nach Situation und Anforderung auch ausgetauscht werden können. Beim situierten Lernen wird mehr Wert auf die Definition der Umgebung gelegt, in der Wissen erlernt wird, wobei hierbei auch soziale Interaktion mit Mitlernern eingeschlossen ist, so daß z.B. der Schüler, der sich ein Gebiet erarbeitet hat, als Tutor auftreten kann und durch das Erklären des Stoffes an seine Mitschüler vertiefte Kenntnisse durch Elaboration und Reflexion erhält⁶.

2.2 Didaktische Umsetzung

Bei all den Theorien zum menschlichen Lernen stellt sich als nächstes die Frage, wie diese in der Praxis angewandt werden können, um mit ihrer Hilfe Stoff zu vermitteln. Die hierzu existierenden Instruktionstheorien werden hier mit etwas Mut zur Lücke nicht näher betrachtet, sondern es soll hier direkt auf das resultierende Instruktionsdesign eingegangen werden, das die Umsetzung konkretisiert.

Sehr einfach am Computer umzusetzen sind die **behaviouristischen** Lernverfahren, die im Drill & Practice-Verfahren abwechselnd den zu vermittelnden Stoff präsentieren, und anschließend das Gelernte abfragen. Die Präsentation des Stoffes kann dabei durch einen (menschlichen) Lehrer oder auch einen Computer geschehen, die Aufbereitung des vermittelten Stoffes kann auf beliebige Medien wie Sprache, Schrift, Ton und Film zurückgreifen. Kurz, alles was der Stand von Technik und Multimedia zu bieten hat. Wird der Stoff nicht durch Menschen vermittelt so kann der Schüler seine Lerngeschwindigkeit selbst bestimmen, was oft als Vorteil multimedialer Lernsysteme gesehen wird. Im Gegensatz zum menschlichen Lehrer können aber Computer nur begrenzt auf Fehler des Schülers eingehen, falls dieser beim Abfragen des gelernten Stoffes Schwierigkeiten aufzeigt⁷.

Mehr und näher auf die individuellen Unterschiede von Schülern einzugehen ist mit Hilfe der behaviouristischen Verfahren nicht möglich, jedoch bieten sich hier durch Anwendung der **kognitivistischen** Verfahren einige Möglichkeiten. Anstatt Instruktionen zu geben wird Wissen dadurch aufgebaut, daß hier Probleme gelöst werden, wobei zum Finden eigener Lösungen ermutigt wird. Die Umgebungen, in denen gelernt wird, können je nach Lernumfeld eher einfach oder sehr komplex sein, bis hin zu simulierten Umgebungen oder Mikrowelten, wobei Papert's "Logo" hier wohl zu den bekanntesten gehört⁸. Je nachdem welche Art von Wissen vermittelt bzw. aufgebaut werden soll sind auch unterschiedliche Aufgabenstellungen anwendbar, etwa um Zusammenhänge

⁵[Bruns and Gajewski, 2002] S. 15

⁶[Mandl et al., 1994] S. 170ff

⁷[Seidel and Lipsmeier, 1989] S. 40f

⁸[Bruns and Gajewski, 2002] S. 32f; [Papert, 1982] S. 152ff

zu erkennen muß das Verhalten des Systems beim Ändern bestimmter Parameter vorausgesagt werden werden, wohingegen zum Erreichen bestimmter Zustände gefragt werden kann welche Parameter zu ändern sind, und zum Lösen von Problemen müssen die richtigen Zustände erkannt und durch Ändern der richtigen Parameter herbeigeführt werden. Als Kontrollinstanz kann dabei entweder Mensch oder Maschine dienen, wobei offensichtlich wird, daß gewisse Anforderungen hinsichtlich Komplexität und Flexibilität hier eine Herausforderung darstellen⁹.

Mit dem Wegfall des Lehrers und sonstiger instruktionsgebender Instanzen im **Konstruktivismus** wird die Umsetzung noch schwieriger. Der Ansatz des Problemlösens bleibt zwar, jedoch muß die Lernumgebung noch wesentlich flexibler auf Aktionen des Schülers reagieren, da weniger bis keine Angaben über die Art der Lösung gemacht werden, und sich dem Schüler damit auch wesentlich mehr Möglichkeiten anbieten, die nicht von vornherein ausgeschlossen werden können bzw. sollten. Zur entsprechenden Präsentation der Inhalte und Umgebung eignen sich Hypermedia-Systeme oder Simulationen, die ein freies Navigieren erlauben, die Unterstützung des Lerner durch das System muß mit Hilfe von (Intelligenter) Tutoriellen Systemen und Adaption an die Eigenarten des Benutzers geschehen¹⁰. Ergänzend zur Unterstützung, die die Lernumgebung bietet, wird Kommunikation als weiteres Mittel eingesetzt, mit dem Fragen an den Lehrer, Tutor oder Mitschüler gestellt werden können. Mitschüler können in diesem Fall auch die Rolle eines Tutors übernehmen, und durch das Erklären des Stoffes, den sie bereits gelernt haben, diesen selbst vertiefen. Die Kommunikation kann dabei "real" oder rechnergestützt geschehen, wobei bekannte Mittel der synchronen bzw. asynchronen Kommunikation zum Einsatz kommen¹¹. Das Problem, daß die Systeme extrem flexibel auf die Benutzer eingehen müssen und dadurch kaum umsetzbar sind hat sich in der Praxis als ausschlaggebend dafür erwiesen, daß sich dieser Ansatz kaum durchgesetzt hat¹².

Beim Umsetzen der **hybriden Ansätze** werden die kommunikativen, tutoriellen und adaptiven Umsetzungen des Konstruktivismus mit den situierten Ansätzen des Kognitivismus vereint, so daß zwar weiterhin der Individualität des Lerner Rechnung getragen wird, dies jedoch nicht dazu führt daß die Systeme aufgrund zu hoher Komplexität nicht mehr realisiert werden können.

2.3 Aspekte der Umsetzung

Im Jahre 1929 philosophierte Edward Thorndike "Wenn es durch ein Wunder technischer Erfindungsgabe gelänge, ein Buch so anzulegen, daß dem Leser erst wenn er getan hat, wozu er auf der ersten Seite angewiesen wurde, die zweite Seite sichtbar wird usw., dann könnte ein großer Teil des heute noch an Personen gebundenen Unterrichts durch Druckwerke ersetzt werden"¹³.

Über diesen Schritt sind wir anerkanntermaßen heute hinweg, jedoch stellt sich nach wie vor die Frage, in wie weit der Computer den Lehrer ersetzen oder ergänzen kann. Das Spektrum reicht hierbei vom klassischen Frontalunterricht wie in der im Kapitel 3 beschriebenen Vorlesung "Systemadministration" über eine Mischung aus Präsenzlehre und virtuellen Komponenten bis hin zu

⁹[Kuyper, 1998] S. 53

¹⁰[Schulmeister, 1997] S. 19ff

¹¹[Bruns and Gajewski, 2002] S. 48ff

¹²[Schulmeister, 2002a] S. 79

¹³[Holland and Skinner, 1961] S. V

rein virtuellen Veranstaltungen, wie sie z.B. von der Virtuellen Hochschule Bayern (VHB) und anderen angeboten wird. Der Lernfortschritt kann dabei unterschiedlich entweder vom Lernenden oder vom Lehrer bzw. der umgebenden Lernumgebung festgesetzt sein.

Je nach zu vermittelndem Inhalt und gewählter Lerntheorie kann die Lernumgebung auch verschiedene methodische und kommunikative Komponenten bereitstellen. Weiterhin ist sie für Bewertung und Feedback an den Lerner verantwortlich, wobei die hier sehr häufig anzutreffenden Multiple Choice Fragen nur den Bereich einfacher, behaviouristischer Fragen abdecken, und wenig auf Kognition und Konzeptverständnis eingegangen wird. Fortschrittlichere Formen wie interaktive Mind Maps, Bilder oder auch die Bewertung einer Situation, welche herbeizuführen Aufgabe des Lernenden war, sind für die Umsetzung kognitivistischer und konstruktivistischer Lerntheorien besser geeignet, jedoch auch im selben Maße seltener zu finden. Der Grund hierfür ist, daß sie erheblich aufwendiger umzusetzen sind, so daß gerade in vielen Umsetzungen, die auf Simulation oder Mikrowelten setzen oft leider gar kein Feedback gegeben wird¹⁴.

Zwei Schlüsse können hieraus gezogen werden: zum einen kann nicht blind jeder klassische Unterricht durch virtuelle Lehre ersetzt werden, zum anderen ist die Umsetzung der rein konstruktivistischen Ansätze in der Praxis kaum realisierbar, so daß sich hier die besprochenen Mischformen anbieten, wobei die genaue Umsetzung immer von Faktoren wie der Zusammensetzung der Lerngruppe, des Stoffgebiets sowie der beabsichtigten Unterrichtsform abhängt.

2.4 Idealer Unterricht

Unter Beachtung der bisher beschriebenen Lerntheorien enthält ein idealer Unterrichtsansatz für ein komplexes Gebiet wie Systemadministration somit die folgenden Schritte: Für eine Sammlung von Aufgaben werden zuerst spontane Lösungsideen gesammelt und diskutiert. Als nächstes werden Lernziele festgelegt und erklärt, warum sie von Bedeutung sind. Die Schritte zum Erreichen der Lernziele werden als nächstes besprochen, und die dafür nötigen Grundlagen vermittelt, womit dann im nächsten Schritt die eigentliche Aufgabe gelöst werden kann. Die verschiedenen Lösungen, die für die gestellten Aufgaben gefunden wurden, werden anschließend aufgestellt und das Gelernte besprochen. Als nächstes werden themenspezifische Aufgaben vorgestellt und bearbeitet, sowie anschließend diskutiert welches Wissen erworben wurde, und auf welchem Weg. Dieses Vorgehen kann sowohl im klassischen Unterricht als auch bei vom Lerner kontrollierten virtuellen Umgebungen eingesetzt werden¹⁵.

Unter Zuhilfenahme einer Reihe von Instrumenten kann weiterhin der Fokus vom Ergebnis des Lernprozesses auf diesen selbst geschoben werden. Diese Instrumente beinhalten leistungsfähige Lernumgebungen zur Förderung der Kreativität, Spiele zur Motivationssteigerung, kognitive Werkzeuge um den Prozeß von Wissen und Verstehen zu fördern und die Darstellung desselben zu unterstützen, sowie weitere Werkzeuge die der Darstellung und Argumentation helfen, und schließlich Programme die die Wiedergabe der mentalen Prozesse des Lerners unterstützen¹⁶.

¹⁴[Schulmeister, 2002b] S. 223

¹⁵[Tulodziecki, 2000] S. 62

¹⁶[Schulmeister, 1997] S. 79

3 Fallstudie

Im vorangegangenen Kapitel wurden verschiedene Lerntheorien, ihre Umsetzung sowie Vor- und Nachteile besprochen. Als nächstes nun soll die am Fachbereich Informatik/Mathematik der FH Regensburg abgehaltene Vorlesung “Systemadministration” als Fallstudie dienen und unter didaktischen Gesichtspunkten besprochen werden.

3.1 Zielpublikum

Die Vorlesung “Systemadministration” wird seit 1994 am Fachbereich Informatik/Mathematik der Fachhochschule Regensburg angeboten. Ursprünglich wurde die Vorlesung als Wahlpflichtfach von Prof. Sauer initiiert, 1999 wurde sie von Hubert Feyrer übernommen und ist seit 2003 Pflichtfach im 5. Semester des Studiengangs (Allgemeine) Informatik. Interessierten Studenten der Studiengänge Wirtschaftsinformatik und technische Informatik steht die Teilnahme und Anerkennung als Wahlpflichtfach offen.

3.2 Inhalt

Die Vorlesung besteht aus zwei Vorlesungen und einer Übung zu je 90 Minuten. Die ca. 40 Studenten werden für die Übungen in zwei Gruppen aufgeteilt, das Lehrpersonal hält die Übung zweimal ab.

Durch ihre Ausbildung und die Inhalte der sonstigen Vorlesungen verfügen die Studenten nur über geringe Unix-Kenntnisse. Grundlagen zu Betriebssystemen und Netzwerken sind vorhanden, jedoch ohne diese routiniert umsetzen und für größere Installationen automatisieren zu können. Aus diesem Grund werden im Rahmen der Vorlesung notgedrungen zuerst die Benutzung des Systems inkl. Befehlen und Konzepten eingeführt und vertieft, bevor im weiteren Verlauf der Vorlesung auf die eigentlichen Tätigkeiten der Systemadministration eingegangen werden kann.

Ziel der Vorlesung und Übungen ist es zum einen, ein generelles Verständnis für Unix zu vermitteln, und dieses Wissen für den Themenbereich Management von Rechner-Clustern inkl. der Teilaspekte Benutzerverwaltung, System Operations und System Startup zu vertiefen. Als Betriebssystem wird primär Solaris eingesetzt, Übungen sind weiterhin auf Systemen unter SuSE Linux und NetBSD/i386 zu absolvieren, um ein breites Systemverständnis zu erlangen und abstrakte Konzepte von konkreten Implementierungen unterscheiden zu können.

Im folgenden soll kurz ein Überblick über den Inhalt der Vorlesung gegeben werden, orientiert an der Aufteilung derselben¹⁷.

Kapitel 1 beginnt mit einem historischer Überblick über die Geschichte von Unix, ausgehend von seiner Entstehung bei AT&T, und geht auf Spinoffs in den Bereichen BSD und Linux sowie kommerzielle Systeme ein, wiederum auf diesen Systemen basieren.

Als praktischen Einstieg diskutiert **Kapitel 2** den klassischen Unix Login-Prozeß sowie die Zu-

¹⁷[Feyrer, 2004c] ‘Inhalt’

sammenhänge der daran beteiligten Prozesse und Konzepte wie Prozesse, Signale und das Benutzen der Manual-Pages.

Kapitel 3 diskutiert gebräuchliche Hilfsprogramme die sowohl standalone als auch für die Shell-Programmierung später von Bedeutung sind, und erläutert Aufgaben und Funktionsweise sowie die darunterliegenden Konzepte des Betriebssystems insbesondere in den Bereichen der Dateisysteme und Zugriffsrechte. Eine Betrachtung regulärer Ausdrücke sowie diverser Programme, die diese Benutzen rundet das Kapitel ab. Die praktischen Übungen der Studenten bestehen darin, sich mit den vorgestellten Programmen und Konzepten auf den zur Verfügung stehenden Übungsrechnern unter den diversen Unix-Derivaten vertraut zu machen.

Nach den "Benutzerprogrammen" werden im **Kapitel 4** Programme und Verfahren aufgezeigt, um Informationen über das System sowie seinen Zustand in verschiedenen Bereichen wie Prozesse, Benutzer, installierte Software, Kernel, Prozeß Accounting, Dateisysteme und Quotas zu erlangen. Besonders wichtig ist auch hier der Vergleich zwischen verschiedenen Betriebssystem-Implementierungen, um allgemeine Konzepte und implementierungsspezifische Details identifizieren zu können. Die praktischen Übungen bestehen aus der Analyse der zur Verfügung stehenden Übungsrechnern. Wo Hardware und/oder Betriebssysteme fehlen werden die entsprechenden Angaben in den Unterlagen detailliert dargestellt, wobei hier die Verfügbarkeit von Systemen z.B. unter SGI/Irix, AIX, HP/UX oder auch Mehrprozessor-Maschinen u.a. von Sun für eine fundiertere Ausbildung sorgen würde.

Nachdem in den vorangegangenen Kapiteln die Umgebung für den täglichen Gebrauch von Benutzern und Administratoren gezeigt wurde stellt sich als nächste Herausforderung beim Management von Rechnerclustern das Problem der Automatisierung. **Kapitel 5** behandelt dazu das Thema Shellprogrammierung, wobei explizit nur auf die klassische Bourne Shell (/bin/sh) eingegangen wird, da diese am weitesten verbreitet ist, und aufgrund ihres Feature-Sets den nötigen Aufgaben gewachsen ist, ohne daß inkompatible bzw. nicht-portable Konstrukte verwendet werden müssen, wie sie leider in vielen anderen Shells zu finden sind. Die Übungen der Studenten beinhalten das Erstellen eigener Scripten zum Vertiefen der vermittelten Kenntnisse.

Um der landläufigen Meinung entgegenzutreten, daß man (Shell-)Programmieren nur und ausschließlich durch Programmieren erlernt werden kann, wird im Anschluß an die Darstellung der Bourne-Shell im **Kapitel 6** mit ein Framework aus Shell-Scripten aufgezeigt, dessen Verständnis durch das Lesen der Scripten erreicht werden soll. Diskutiert wird das Starten des Unix-Systems, mit seinen verschiedenen Varianten wie dem klassischen Unix-rc-System, dem System V Init-Konzept sowie den verschiedenen Arten die Startreihenfolge der Systemdienste festzulegen. Da verschiedene Systeme hier unterschiedliche Konzepte unterschiedlich realisieren, jedoch im allgemeinen als Shell-Scripten wird hier gleichzeitig eine Anwendung des bisher vermittelten Stoffes mit einer eingehenderen Auseinandersetzung mit neuem Stoff gezeigt. Die praktischen Übungen müssen dabei leider auf der Ebene der Analyse der vorhandenen Systeme verbleiben, da nicht die nötigen Ressourcen vorhanden sind, um die Rechner nach Änderungen am System neu zu installieren.

Ähnlich ist die Situation beim Stoff von **Kapitel 7**, Networking. Die Studenten besitzen Grundkenntnisse im Bereich Netzwerke und TCP/IP, sind jedoch mit konkreten Implementierungen sowie der Konfiguration nicht vertraut. Nach einer kurzen Wiederholung der Grundlagen von TCP/IP wird anschließend erklärt wie die grundsätzliche Netzwerkkonfiguration zu bewerkstell-

gen ist, angefangen bei Netzwerk-Interfaces über Routing bis hin zu den Namensdiensten. Ausgehend von dieser Basis werden einige wichtige Netzwerk-Dienste näher vorgestellt, wobei besonderes Augenmerk auf die Public-Key-Authentifizierung der SSH sowie den Setup von Clients und Servern sowohl für das Network File System (NFS) als auch das Network Information System (NIS) gelegt wird. Kritiker an diesen Diensten seien dabei auf den Multi-Plattform-Anspruch der Vorlesung sowie an die Tatsache verwiesen, daß bisher keine verbreiteten Alternativen zu diesen Diensten existieren. Die praktischen Übungen der Studenten in diesem Bereich beschränken sich wiederum auf die Analyse der bestehenden Systeme sowie das Umsetzen der Public-Key-Authentifizierung, die grundsätzliche Netzwerk-Konfiguration sowie die Dienste NIS und NFS können nicht praktisch erforscht werden, da hierzu wiederum nicht die nötigen Ressourcen vorhanden sind um die Rechner nach den Übungen wieder in einen definierten Zustand zu versetzen, und weiterhin kein abgekoppeltes Übungsnetz existiert, in dem Studenten den Produktivbetrieb nicht stören können.

Wesentlich mehr praktische Übungen sind beim X Window System im **Kapitel 8** möglich. Nach den Grundlagen wie X Client und Server und der Addressierung entfernter Displays wird das Umleiten der Anzeige von Anwendungen "manuell" und mit Hilfe von SSH erklärt sowie die Zugriffskontrolle auf das X-Display mittels xhost und X Authority. Im Anschluß folgen die verschiedenen Arten X zu starten sowie die Erklärung der Funktion und Konfiguration von Window Managern. Eine Demonstration der Komponenten des KDE Desktops rundet den Überblick ab. Die Studenten sind in der Übung aufgefordert, den vermittelten Stoff zu vertiefen, wobei besonderer Wert auf das Umleiten der Ausgabe sowie die Konfiguration eines Window Managern gelegt wird.

Zum nächsten Thema Security existiert an der FH Regensburg zwar eine eigene Vorlesung, jedoch vermittelt diese allgemeine Sicherheitskonzepte und -Methoden, die konkrete Umsetzung dieser wird in **Kapitel 9** diskutiert. Aus dem breiten Spektrum werden die Themen Host Security und Network Security sowie das Erlangen von Informationen herausgegriffen. Für eine Reihe von Problemkreisen wie Datenmanipulation, dem unerlaubten Benutzen von Ressourcen und dem unerlaubten Erlangen von Zugriffsrechten wird jeweils die bestehende Problematik erklärt, gefolgt von Methoden zur Erkennung sowie Gegenmaßnahmen, präventiv und post-mortem. Die praktischen Übungen umfassen die Analyse der bestehenden Systeme auf mögliche Sicherheitsprobleme sowie das experimentieren mit Buffer Overflows. Die Studenten werden angehalten, gefundene Probleme unverzüglich dem Übungsleiter zu melden.

Nachdem mit den Themen X Window System und Security zwei Themen angesprochen wurden, die nicht direkt dem Vorlesungsziel des Cluster Managements dienen, soll mit den beiden folgenden Kapiteln der letzte Pfeiler dieses Themengebiets abgedeckt werden. **Kapitel 10** liefert eine Einführung in die Programmiersprache perl. Hintergrund ist hier wiederum die Automatisierung wiederkehrender Routineaufgaben wie der Auswertung bestehender Protokolldateien sowie das Benutzen von Schnittstellen zum System, die in der Bourne Shell nur unzureichend vorhanden sind. Die Spracheinführung ist allgemein gehalten, die Übungen entsprechen dem, wobei das Ermitteln und Darstellen von Systeminformationen betont wird.

Im **Kapitel 11** wird das Thema Benutzerverwaltung aufgegriffen. Die mittlerweile bekannten Schritte zum manuellen Anlegen von Kennungen werden wiederholt, und Tools um dies zu automatisieren vorgestellt. Den Kern des Kapitels bildet die Vorstellung der Benutzerverwaltung des Fachbereichs Informatik/Mathematik, der basierend auf dem Auszug des FH-weiten NDS-

Datenbank Kennungen sowohl im Unix-NIS/NFS-Verbund als auch auf Windows 2000 anlegt. Die verwendeten Scripten sind größtenteils in perl geschrieben, so daß die im vorhergehenden Kapitel gelernten Kenntnisse hier durch das Lesen bestehenden Codes vertieft, und der Vorgang des Anlegens und Löschens von Benutzerkennungen praktisch nachvollzogen werden kann. Die praktischen Übungen der Studenten beschränken sich hier wiederum auf die Analyse verschiedener Systeme und ihrer Werkzeuge zur Benutzerverwaltung, praktische Übungen müssen wiederum entfallen, da keine Systemverwalter-Rechte herausgegeben werden können.

Mit dem vorletzten **Kapitel 12** “Software-Management” wird die Software-Verwaltung im Unix-Bereich besprochen, und insbesondere die Trennung zwischen Betriebssystem und Anwendungen sowie den sich daraus ergebenden Software- und System-Architekturen. Nach den Rahmenbedingungen wie Abhängigkeiten und Verzeichnisstrukturen wird die Benutzung von vorcompilierten Binärpaketen auf verschiedenen Systemen näher besprochen, wobei der Fokus hier auf das Umsetzen der eigenen Verzeichnisstruktur und die Möglichkeit, auf Systemverwalter-Rechte zu verzichten, gesetzt ist. Die praktischen Übungen der Studenten profitieren von diesen Punkten, da sie so selbst Software installieren können, und auch in der Praxis ist Software-Installation mit Systemverwalter-Rechten nicht erstrebenswert. Die Problematik, Open Source Software aus dem Quellcode zu installieren wird in der Vorlesung “Systemadministration” explizit ausgeklammert, da dieses Thema im Rahmen der Wahlpflicht-Vorlesung “Open Source” eingehender besprochen wird¹⁸.

Kapitel 13 beschäftigt zum Abschluß der Vorlesung mit dem Thema Datensicherung. Es werden verschiedene Arten der Sicherung wie einzelne Dateien, Verzeichnisbäume, ganze Dateisysteme bzw. Plattenimages mit den zugehörigen Werkzeugen besprochen und auch das Thema Datenkompression angesprochen sowie kommerzielle Backup-Systeme angesprochen. Die praktischen Übungen beschränken sich mangels Zugriffsrechte und Backup-Hardware eher auf die Sicherungsstrategien für persönliche Daten der Studenten.

3.3 Layout

Bei genauer Betrachtung der in der Vorlesung behandelten Themen fällt auf, daß diese teilweise aufeinander aufbauen. Wenn man diese Abhängigkeiten aufstellt, so wird dies klar, z.B. müssen für die Shellprogrammierung (Kapitel 5) grundlegende Benutzerkommandos (Kapitel 3) und Befehle zur Abfrage des Systemzustandes (Kapitel 4) bekannt sein, bevor das Starten des Systems (Kapitel 6) diskutiert wird. Für die “großen” Themen Benutzerverwaltung (Kapitel 11) und Softwaremanagement (Kapitel 12) können ähnliche Voraussetzungen definiert werden, so daß sich der in Abbildung 1 gezeigte Graph für das Ziel des Cluster Managements ergibt.

Der Graph zeigt, daß sich die Themen grob in drei große Themengruppen einteilen lassen: Benutzerverwaltung, der Betrieb des Systems sowie das Starten des Systems mit seinen zugehörigen Diensten. Abbildung 2 verdeutlicht dies.

Die Betrachtung der einzelnen inhaltlichen Gruppen zeigt, daß hier wiederum eine Unterteilung in Schwierigkeitsgrade in Form von Grundlagen, fortgeschrittenen Themen und den eigentlichen Lernzielen der jeweiligen Gruppe aufgesplittet werden kann, wie in Abbildung 3 gezeigt.

¹⁸[Feyrer, 2004b] ‘Inhalt’

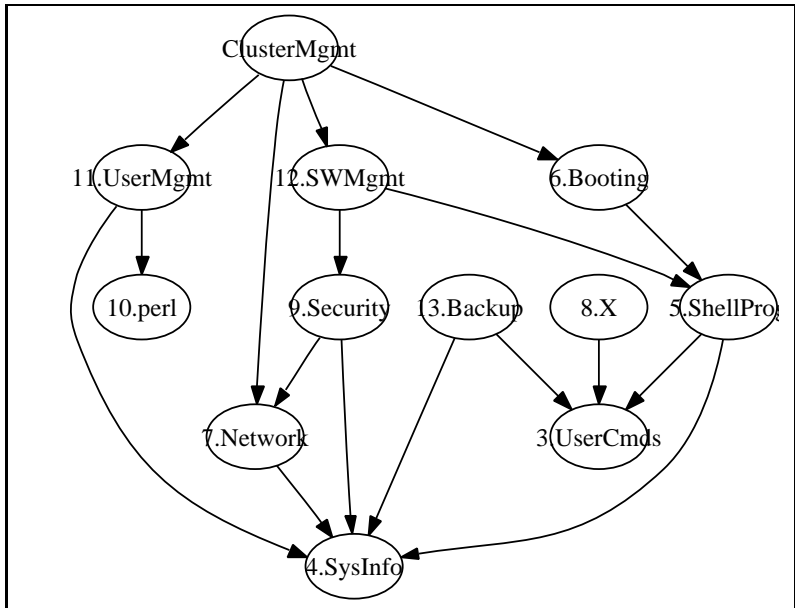


Abbildung 1: Layout

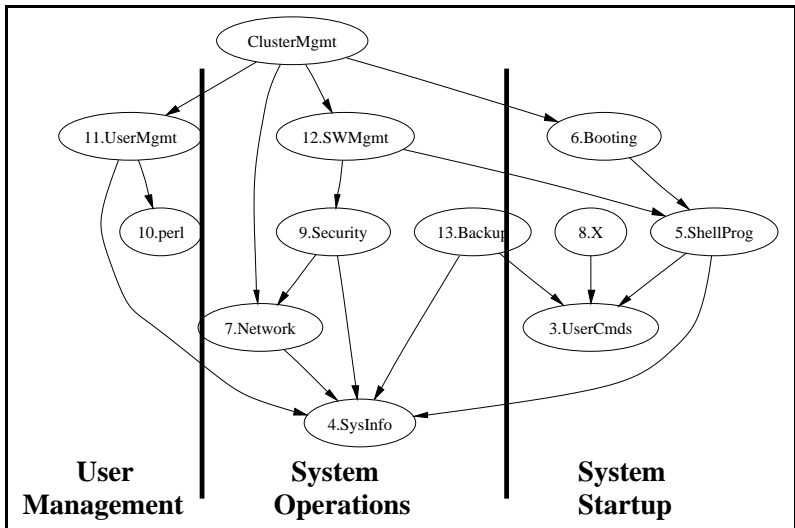


Abbildung 2: Themengruppen

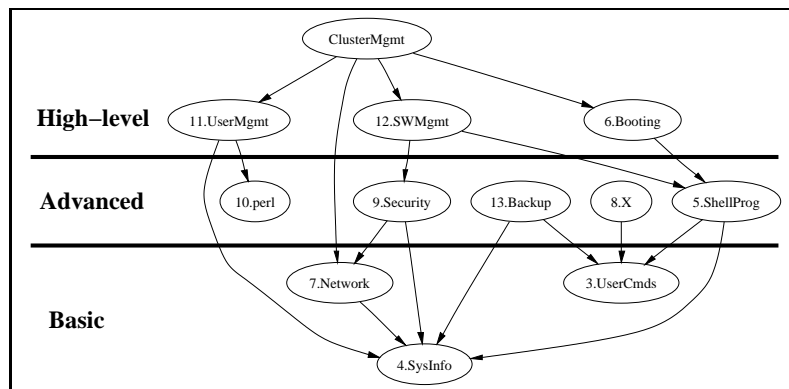


Abbildung 3: Schwierigkeitsgrade

Da die einzelnen Themengruppen aufgrund ihrer Komplexität nicht geschlossen behandelt werden können wurden sie wie aufgezeigt in die verschiedenen Schwierigkeitsgrade aufgeteilt. Zusammenfallende Themen wurden weiter gegliedert, so daß sich die in Abschnitt 3.2 gezeigte Kapitelstruktur ergibt.

3.4 Didaktische Instrumente

Zur Umsetzung der Vorlesung wie sie in den vergangenen Abschnitten vorgestellt wurde werden eine Reihe von didaktischen Instrumenten eingesetzt, um die einzelnen Lernziele zu erreichen.

Die **Vorlesung** ist mit vier Semesterwochenstunden zentraler Bestandteil. Zur Präsentation des Unterrichtsmaterials werden Beamer & Laptop eingesetzt, einzelnen Beispiele werden auch an der Tafel entwickelt, die **Vorlesungsunterlagen** sind online verfügbar¹⁹. Die Vorlesungsunterlagen enthalten viele vollständige **Beispiele**, um ein Studium auch ohne Rechner zu erlauben, unterstützt wird dies durch **Demonstrationen** während der Vorlesung, wobei hier Wert auf das Darlegen des Zieles, dem Weg der Umsetzung und der anschließenden Besprechung des Ergebnisses gelegt wird. Neben der passiven, behavioristischen Wissensvermittlung wird ein vergleichbares Vorgehen auch bei der **Analyse bestehender Systeme** unter den verschiedenen Aspekten der Vorlesung angestrebt. Dabei wird Wert auf die Analyse verschiedener Unix-Systeme gelegt, um kognitivistische Konzepte in Form von allgemeine und portable Konzepte von konkreten, unportablen Implementierungen unterscheiden zu lernen. Um Studenten weiterhin auch Hands-on Wissen zu vermitteln werden **praktische Übungen** durchgeführt, wobei die Studenten zu den jeweiligen Themen Aufgaben gestellt bekommen. Diese Aufgaben sind im Rahmen der **betreuten Übung** zu lösen sind, bei denen die Studenten Fragen zum Thema und den Aufgaben stellen können.

Neben diesen Instrumenten wären noch eine Reihe weiterer Instrumente zur qualitativen Verbesserung der Vorlesung denkbar. Angefangen von Musterlösungen an denen Studenten konkrete Lösungswege ersehen könnten über den Zugriff auf mehr verschiedene Rechner mit verschiede-

¹⁹[Feyrer, 2004c] ‘Inhalt’

nen Betriebssystemen wie SGI-Rechner unter Irix, SPARC- und AMD-basierende Sun Workstations und -Server sowie diverse Linux-Distributionen könnten den zu erwerbenden Blickwinkel erweitern. Der zusätzliche Zugriff mit Systemverwalter-Rechten wäre sehr wünschenswert, da v.a. die fortgeschrittenen Übungen nicht ohne diesem praktisch umgesetzt werden können. Mit dem Zugriff auf Systemebene wären auch weitere Aufgaben wie die Installation verschiedener Betriebssysteme inkl. deren Konfiguration zur Anpassung an bestimmte Umgebungen, das Aufsetzen einzelner Dienste wie EMail, POP, IMAP, Spam-Filtern, DNS, DHCP, FTP, SSH, Samba, NFS, NIS und vielen anderen sowie auch die Einbindung von Hardware wie z.B. in den Bereichen Netzwerk und Backup nötig. Problem bei Vergabe von Systemverwalter-Rechten ist, daß zum einen auch ein eigenes Netzwerk-Segment benötigt wird um den Produktivbetrieb nicht zu stören, zum anderen die Rechner nach der Übung in einem undefinierten Zustand sind und für den sonstigen Laborbetrieb neu installiert werden müßten. Die Betreuer der Übungsrechner sind hierzu jedoch aufgrund des nötigen Zeitaufwands nicht in der Lage.

Nachdem die zugrundeliegenden Konzepte für bestimmte Konfigurationsaufgaben wie die nötigen Befehle und zu ändernden Dateien auf dem jeweiligen Betriebssystem bekannt sind könnten auf diesen aufbauend unterstützende Werkzeuge wie SuSE "yast", die Solaris Management Console oder plattformübergreifende Werkzeuge eingeführt werden, um die Umsetzung dieser dann bekannten Aufgaben zu erleichtern. Ein sofortiges und ausschließliches Einsetzen solcher Werkzeuge wird jedoch als negativ erachtet, da die darunterliegenden Konzepte und Eigenarten der jeweiligen Systeme damit nicht oder nur unzureichend erlernt werden können.

Um bestimmte Setup- oder Troubleshooting-Situationen üben zu können wäre es wünschenswert, ein bestimmtes System in einem bestimmten Zustand allen Studenten zur Verfügung stellen zu können, um diese von dieser Situation ausgehend arbeiten lassen zu können. Neben dem Mangel an Hardware ist hier auch der Aufwand zum Aufsetzen des Systems in den bestimmten Zustand zu beachten.

Letztlich wäre noch die Überlegung anzustellen, ob die momentan als Präsenzvorlesung existierende Veranstaltung alternativ als rein virtuelle Veranstaltung angeboten werden soll, bei der die Studenten sich den Stoff anhand der Online-Unterlagen selbst beibringen. Das Durchführen der Übungen und dabei die Unterstützung durch den Dozenten wären hier zu meisternde Herausforderungen.

4 Analyse der aktuellen Situation & Ausblick

Basierend auf den in Kapitel 2 dargelegten didaktischen Grundlagen und der in Kapitel 3 gegebenen Fallstudie soll hier die aktuelle Situation der Ausbildung im Bereich Systemverwaltung analysiert werden.

Mit dem in Kapitel 2.4 gegebenen Vorgehen für idealen Unterricht und Werkzeuge kann eine Verbesserung der Orientierung für Studenten vorgeschlagen werden: Zum einen kann das Gesamtziel der Vorlesung klargemacht werden, Allgemeines Verständnis für Unix und das Management von Rechner-Clustern; Zum anderen kann der Weg dorthin durch das Aufzeigen der Vorlesungsgliederung sowie eine Inhaltsübersicht am Anfang jedes Kapitels zu einem gesteigerten Überblick führen. Diese Änderungen können in künftigen Inkarnationen der Vorlesung "Systemadministra-

tion” leicht eingebunden werden.

Ein anderes Problem ist schwieriger zu Lösen: bei den fortgeschrittenen Themengebieten ist ein praktisches Üben unerlässlich, wie aus der Beschreibung der aktuellen Vorlesung in den Kapitel 3.2 und 3.3 sowie der real eingesetzten als auch insbesondere den gewünschten didaktischen Instrumente aus Kapitel 3.4 ersichtlich ist. Merrill unterstreicht diese Vorgehen mit seiner Aussage daß “much new scientific knowledge is dynamic in character and cannot be understood without a more active prepresentation and student involvement”²⁰, und die Ansätze zu situiertem Lernen und verwandte in Kapitel 2.1 vorgestellte kognitive Konzepte bestätigen dies.

Die Grundlagenkapitel des Kurses können ohne praktische Übungen und mit rein behavioristischen Lernverfahren erlernt werden, jedoch verschiebt sich der Fokus bei Fortschreiten des Kurses auf kognitivistische und konstruktivistische Konzepte, die durch die praktische Auseinandersetzung mit dem Stoff verbunden sind. Dieser Übergang von behavioristischen zu kognitivistischen bzw. konstruktivistischen Lerntheorien zeigt auch, daß die in Kapitel 2.1 genannten Mischformen nötig sind. Leider ist es momentan nicht möglich, viele der Übungen und Themen praktisch umzusetzen, wie bereits in Kapitel 3.2 dargelegt wurde. Für die praktische Umsetzung v.a. mit Systemprivilegien fehlt momentan Personal, Hardware und Prozeduren zum schnellen installieren und wiederherstellen entsprechender Übungsrechner. Eine mögliche Lösung dieses Problems wäre ein virtuelles Labor, das die Umsetzung dieser Anforderungen erlaubt.

Aspekte eines solchen virtuellen Labors wären automatische Installation von Übungsrechnern inkl. Neuinstallation am Ende einer Übung, Zugriff der Studenten auf das System über das Internet und Auswertung der Übungsleistung und Bereitstellen von Feedback für den Einsatz im Rahmen der bestehenden Vorlesung. Weiterführende Themen wären Unterstützung der Benutzer durch eine tutorielle Komponente v.a. in Hinblick auf rein virtuelle Lehrveranstaltungen bei denen der Kontakt zwischen Student und Lehrer im Gegensatz zu Präsenzveranstaltungen nicht direkt gegeben ist, und Adaption um sowohl erfahrenen Studenten als auch Einsteigern gleichzeitig optimale Unterstützung geben zu können.

Am Fachbereich Informatik/Mathematik der Fachhochschule Regensburg wurde im Rahmen des Projektes “Unix Cluster Setup” mit Mitteln des Hochschul- und Wissenschaftsprogramms (HWP) des BMBF das Virtuelle Unix Labor entwickelt, das einen Basisumfang dieser Forderungen erfüllt. Das System wurde im Sommersemester 2004 erfolgreich getestet und ist in [Feyrer, 2004d] und [Feyrer, 2004a] näher beschrieben. Die informationswissenschaftlichen Aspekte wie die Auswertung der Übungsleistung, Feedback, tutorielle Unterstützung und Benutzeradaption werden im Rahmen der Dissertation “Sysadmin Education in the Virtual Unix-Lab” am Lehrstuhl für Informationswissenschaft der Universität Regensburg eingehender behandelt.

Für eine Weiterentwicklung des Systems fehlen momentan durch das Auslaufen der HWP-Förderung jegliche Personalmittel. Interessierte Sponsoren mögen sich bitte an den Autor wenden!

²⁰[Merrill et al., 1991] S. 7

Literatur

- [Bruns and Gajewski, 2002] Bruns, B. and Gajewski, P. (2002). *Multimediales Lernen im Netz – Leitfaden für Entscheider und Planer*. Springer Verlag, Heidelberg, Germany.
- [Feyrer, 2004a] Feyrer, H. (2004a). An Introduction to Sysadmin Training in the Virtual Unix Lab. In *EuroBSDCon 2004 Proceedigns*. Available from: <http://www.feyrer.de/Texts/Own/eurobsdcon2004-vulab-paper.pdf>.
- [Feyrer, 2004b] Feyrer, H. Open Source [online]. (2004) [cited 2004-11-23]. Available from: <http://www.feyrer.de/OS/>.
- [Feyrer, 2004c] Feyrer, H. Systemadministration unter Unix [online]. (2004) [cited 2004-11-18]. Available from: <http://www.feyrer.de/SA/>.
- [Feyrer, 2004d] Feyrer, H. Virtuelles Unix Labor [online]. (2004) [cited 2004-12-18]. Available from: <http://www.feyrer.de/vulab/>.
- [Holland and Skinner, 1961] Holland, J. G. and Skinner, B. F. (1961). *The analysis of behaviour*. McGraw Hill, New York, NY, USA.
- [Kerres, 1998] Kerres, M., editor (1998). *Multimediale und telemediale Lernumgebungen: Konzeption und Entwicklung*. Oldenbourg Verlag, M'unchen, Germany.
- [Kuyper, 1998] Kuyper, M. (1998). *Knowledge Engineering for Usability*. PhD thesis, University of Amsterdam, Roetersstraat 15, 1018 WB Amsterdam, Netherlands.
- [Mandl et al., 1994] Mandl, H., Gruber, H., and Renkl, A. (1994). Situiertes Lernen in multimedialen Lernumgebungen. In Issing, L. J. and Klimsa, P., editors, *Information und Lernen mit Multimedia*, pages 167–178. Psychologie Verlags Union, Weinheim, Germany.
- [Merrill et al., 1991] Merrill, D., Li, Z., and Jones, M. (1991). Second Generation Instructional Design (ID₂). *Educational Technology*, 30(1):7–11. Available from: <http://www.id2.usu.edu/Papers/ID1&ID2.PDF> [cited 2004-11-12].
- [Papert, 1982] Papert, S. (1982). *Mindstorms: Kinder, Computer und neues Lernen*. Birkh'ausser Verlag, Basel, Switzerland.
- [Schulmeister, 1997] Schulmeister, R. (1997). *Grundlagen hypermedialer Lernsysteme*. Oldenbourg Verlag, M'unchen, Germany.
- [Schulmeister, 2002a] Schulmeister, R. (2002a). *Grundlagen hypermedialer Lernsysteme*. Oldenbourg Verlag, M'unchen, Germany.
- [Schulmeister, 2002b] Schulmeister, R. (2002b). *Lernplattformen für das virtuelle Lernen*. Oldenbourg Verlag, M'unchen, Germany.
- [Seidel and Lipsmeier, 1989] Seidel, C. and Lipsmeier, A. (1989). *Computerunterstütztes Lernen – Entwicklungen, Möglichkeiten, Perspektiven*. Verlag für Angewandte Psychologie, Stuttgart, Germany.

- [Shannon and Weaver, 1949] Shannon, C. E. and Weaver, W. (1949). *The mathematical theory of communication*. University of Illinois Press, Urbana, IL, USA.
- [Tulodziecki, 2000] Tulodziecki, G. (2000). Computerunterstütztes Lernen aus medienpädagogischer Sicht. In Kammerl, R., editor, *Computerunterstütztes Lernen*, pages 53–72. Oldenbourg Verlag, München, Germany.