

Vom Problem zur Evaluation - Folgerungen für die Entwicklung computerunterstützter Lernumgebungen

Christoph Richter, Heidrun Allert, Wolfgang Nejdil
Learning Lab Lower Saxony, Universität Hannover

Zusammenfassung

Der wahrgenommene Mangel an entwicklungsbegleitenden Evaluationsansätzen stellt den Ausgangspunkt für die Entwicklung eines eigenen Evaluationskonzeptes dar, das Entwicklern computerunterstützter Lernumgebungen entscheidungsrelevante Informationen liefern soll. Computerunterstützte Lernumgebungen werden hierbei als komplexe, dynamische und intransparente Realitätsausschnitte aufgefasst. Anhand der Unterscheidung von kontextunabhängigen, kontextabhängigen und situierten Problemen wird sowohl die Eignung unterschiedlicher Vorgehensmodelle wie auch Anforderungen an Evaluationskonzepte diskutiert. Für den Bereich der situierten Probleme, zu denen auch die Gestaltung computerunterstützter Lernumgebungen zu zählen ist, wird ein integratives Evaluationskonzept skizziert.

1 Einleitung / Motivation

Evaluation wird immer stärker als ein integraler Bestandteil der Gestaltung und Entwicklung computerunterstützter Lernumgebungen verstanden. Aus Sicht der Entwickler sind formative, also entwicklungsbegleitende Evaluationsansätze, deren Ergebnisse, Empfehlungen für die weitere Entwicklung liefern können, besonders wichtig. Demgegenüber besteht jedoch ein Mangel an entwicklungsbegleitenden Evaluationsansätzen, die den spezifischen Anforderungen bei der Gestaltung computerunterstützter Lernumgebungen gerecht werden (vgl. z.B. Keil-Slawik, 1999).

Die in der Praxis häufig anzutreffende Skepsis gegenüber dem Nutzen einer Evaluation eigener Projekte ist möglicherweise darauf zurückzuführen, dass viele Evaluationsansätze keine handlungsrelevanten Ergebnisse erbringen, oder die Ergebnisse erst zum Abschluss des Projektes vorliegen, so dass sie folgenlos bleiben müssen. Von zentraler Bedeutung sind deshalb Evaluationsansätze, die rechtzeitig entscheidungsrelevante Informationen bereitstellen können.

Evaluationsansätze, die entwicklungsbegleitend eingesetzt werden, müssen den Merkmalen des Gestaltungsgegenstandes gerecht werden. Deshalb ist es wichtig, die Eignung unterschiedlicher Evaluationsansätze in Abhängigkeit von den Merkmalen des Gestaltungsgegenstandes zu betrachten.

Im Rahmen dieses Papers werden zunächst allgemeine Merkmale des Gestaltungsgegenstandes ‚computerunterstützte Lernumgebungen‘ skizziert. Anschließend wird ein generelles Klassifikationssystem für Probleme vorgestellt, in das sich die Entwicklung computerunterstützter Lernumgebungen einordnen lässt. Anhand dieses Klassifikationssystems für Probleme wird die Eignung verschiedener Vorgehensmodelle und Anforderungen an entsprechende

Evaluationskonzepte diskutiert. Abschließend wird ein integriertes Gestaltungs- und Evaluationskonzept für computerunterstützte Lernumgebungen in seinen Grundzügen skizziert.

Die hier dargestellten Konzepte und Argumente entstanden vor dem Hintergrund unserer Erfahrungen bei der Beratung unterschiedlichster Projekte zu Fragen der Evaluation innerhalb des Learning Lab Lower Saxony [L3S]. Das Learning Lab Lower Saxony ist eine multidisziplinäre Forschungseinrichtung, die sich mit der Entwicklung innovativer Lerntechnologien beschäftigt und Mitglied des Wallenberg Global Learning Networks ist. Innerhalb des L3S besteht die Hauptaufgabe des Evaluationsteams in der Beratung und Koordination von Evaluationsvorhaben der aktuellen Projekte in Zusammenarbeit mit den Projektmitarbeitern.

2 Merkmale des Gestaltungsgegenstands „computerunterstützte Lernumgebungen“

Unter computerunterstützten Lernumgebungen werden hier alle Situationen gefasst, in denen intentionales Lernen und / oder Lehren durch den Einsatz von Computern unterstützt werden soll. Der Gestaltungsgegenstand beschränkt sich somit nicht auf die Entwicklung technischer Artefakte, sondern umfasst auch alle nicht-technischen Aspekte einer Lernsituation. Die Bandbreite der computerunterstützten Lernumgebungen reicht von teletutoriell betreuten Fernlaboren, über hybride Lerngemeinschaften und adaptive Hypermedia-Lernsysteme bis hin zu virtuellen Universitäten. Allen gemein ist, dass sie soziotechnische Systeme darstellen, die als komplexe, dynamische und intransparente Realitätsausschnitte im Sinne Dörners (1993) anzusehen sind. Die Komplexität computerunterstützter Lernumgebungen ergibt sich aus den vielfältigen Wechselwirkungen zwischen sozialen, ökonomischen, politischen, technischen sowie ökologischen Bedingungen und didaktischen Konzepten und personalen Merkmalen der Akteure, die Einfluss auf die Lernumgebung und das in ihr eingesetzte technische Artefakt haben (s.a. Pfister & Wessner, 2000). Die Gestaltung von Lernumgebungen sowie der in ihnen eingesetzten Artefakte muss dieser Komplexität Rechnung tragen und kann sich nicht isoliert auf einzelne Komponenten beschränken.

Neben diesen strukturellen Eigenschaften sind computerunterstützte Lernumgebungen verschiedenen dynamischen Prozessen ausgesetzt. Einerseits stellt z.B. die Gestaltung von technischen Artefakten selbst einen Entwicklungsprozess dar, der menschliche Handlungsmöglichkeiten verändert und damit sowohl neue Möglichkeiten wie auch Probleme schafft (Carroll, 2000, p. 46). Andererseits werden computerunterstützte Lernumgebungen auch durch eine Vielzahl externer Entwicklungsprozesse beeinflusst, die unabhängig von den Gestaltern ablaufen. Seitens der Lerner kann steigende Medienkompetenz die Nutzung technischer Funktionen beeinflussen. Urheberrechtliche Bestimmungen können den Austausch von Lernmaterialien beeinflussen. Ökonomische Bedingungen können Einfluss auf die Kostenpflichtigkeit von Bildungsangeboten nehmen. Sich wandelnde Anforderungen des Arbeitsmarktes können neue Bildungsinhalte nötig machen. Neben den Wirkungszusammenhängen in dem Gestaltungsfeld sind somit Kenntnisse über Entwicklungstrends von zentraler Bedeutung für den Gestaltungsprozess (vgl. Dörner, 1993).

Ferner sind nicht alle relevanten Variablen direkt messbar, aufgrund der hohen Komplexität in der Regel sind nicht alle Wirkungszusammenhänge erkennbar und aufgrund sehr langsamer oder

möglichen Operationen. Sie haben eine richtige Lösung. Der Problemlöseprozess basiert auf logischen und algorithmische Prozeduren (vgl. Kitchener, 1983). Die Korrektheit der Lösung bezieht sich ausschließlich auf die Spezifikation. Das Problem steht in keiner kausalen Beziehung zur realen Welt (vgl. Lehman, 1980). Kontextunabhängige Probleme sind zum Beispiel die Türme von Hanoi, das Acht-Damen Problem, das Neun Punkte Problem.

Nach Lehman (1980) stellen die sogenannten S- („spezifikationsbasierte“) Programme Lösungen für kontextunabhängige Probleme dar. Kontextunabhängige Probleme werden oft auch als Puzzle-Probleme bezeichnet und gehören zu den gut-strukturierten Problemen (vgl. Jonassen, 1997). Dementsprechend wäre zum Beispiel auch die Entwicklung eines Suchalgorithmus für eine bestimmte Art von Datenbank ein kontextunabhängiges Problem, auch wenn im Endeffekt mit Hilfe des entsprechenden Suchalgorithmus konkrete Lernmaterialien in einer konkreten Datenbank zu einem konkreten Zweck gesucht werden sollen. Gleiches gilt für die Entwicklung eines Programms, das Zeichenketten auf einem Bildschirm ausgegeben kann.

3.2 Kontextabhängige Probleme

Kontextabhängige Probleme sind solche Probleme, die zwar präzise definiert werden können, bei denen die Güte der Lösung jedoch von Merkmalen der realen Umwelt abhängt (vgl. Lehman, 1980). Bei kontextabhängigen Problemen kann sowohl Ausgangs- als auch Zielzustand festgestellt werden. Ferner sind auch die möglichen Operatoren, die jeweils zu vorhersagbaren Ergebnissen führen benennbar. Kontextabhängige Probleme haben konvergente Lösungen deren Güte durch die Merkmale der realen Umwelt bestimmbar sind. Kontextabhängige Probleme setzen für ihre Lösung Modelle des jeweiligen Kontextes voraus (s.a. Lehman, 1980). In der Regel sind nur ausgewählte Teilaspekte der Realität einbezogen (s.a. Lehman, 1980). Kontextabhängige Probleme sind z.B. das Schachspiel (als Zweipersonenspiel), Wetterprognosen, oder die Entwicklung psychometrischer Verfahren.

Kontextabhängige Probleme erfordern nach Lehman (1980) die sogenannten P- („problembasierten“) Programme. Kontextabhängige Probleme gehören ebenfalls zu den gut-strukturierten Problemen, wenn geeignete Modelle der Realität für die Probleme verfügbar sind. Ob ein kontextabhängiges Problem allerdings gut oder schlecht definiert ist, ist hingegen eher als ein Merkmal der Repräsentation des Problems beim Problemlöser anzusehen und keine Eigenschaft des Problems an sich. Bezogen auf das Beispiel einer Datenbank, die Lernmaterialien enthält könnte sich folgendes kontextabhängiges Problem stellen: Wie kann sichergestellt werden, dass ein Anwender die Lernmaterialien angeboten bekommt, die seinen Anforderungen entsprechen. Die automatisierte Vergabe von Terminen, an denen Lernergruppen auf ein Fernlabor zugreifen können wäre ein weiteres Beispiel. Hierbei sind z.B. überschneidende Termine zu vermeiden und zeitliche Rahmenvorgaben der verschiedenen Lernergruppen zu berücksichtigen¹.

3.3 Situierete Probleme

Situierete Probleme sind solche Probleme, deren Lösung eine Veränderung der realen Umwelt erfordert und somit die Eigenschaften des Problems verändert, das gelöst werden soll. Ausgangs-

¹ Die automatisierte Vergabe von Terminen kann allerdings auch als ein situieretes Problem aufgefasst werden kann, da sie u.U. den Prozess der Terminplanung aus Sicht der Lernergruppe verändert.

und/oder Zielzustand sind bei situierten Problemen unbekannt oder unklar. Ebenso ist die Menge möglicher Operationen oftmals unbekannt und/oder die Effekte einzelner Operationen sind nicht genau vorhersagbar. Für situierte Probleme gibt es in der Regel divergente Lösungsmöglichkeiten, deren Qualität nicht objektiv bestimmbar ist. Jede Lösung hat spezifische Vor- und Nachteile. Die Lösung situiert Probleme erfordert meist die integrative Betrachtung verschiedener Realitätsaspekte. Zu den situierten Problemen zählen zum Beispiel die meisten politischen oder soziologischen Probleme. Dem Konzept der situierten Probleme können die sogenannten E- („eingebettete“) Programme von Lehman (1980) zugeordnet werden. Der Begriff der situierten Probleme wird zum Teil synonym mit dem der schlecht-strukturierten Probleme verwendet (Jonassen, 1997), obwohl beide Konzepte unterschiedliche Schwerpunkte setzen. Die Integration einer Datenbank für Lernmaterialien in ein konkretes Seminar würde somit einem situierten Problem entsprechen. Ein weiteres Beispiel für ein situiertes Problem ist die Entwicklung tele-tutorierlicher Konzepte zur Unterstützung von Lernern bei der Arbeit mit Fernlaboren.

In der psychologischen Problemlöseforschung wurde oftmals die Ansicht vertreten, dass schlechtstrukturierte Probleme dadurch gelöst werden können, in dem man sie in gut strukturierte Probleme überführt (z.B. Zimbardo, 1995). Demgegenüber gibt es eine Reihe neuerer Ansätze, die betonen, dass schlecht strukturierte Probleme qualitativ andere Prozesse vom Problemlöser erfordern. Der obige Klassifikationsvorschlag sollte darüber hinaus auch verdeutlichen, dass es generell qualitative Unterschiede zwischen den verschiedenen Problemen gibt, so dass situierte Probleme nicht auf kontextabhängige Probleme reduziert werden können, ohne dass wichtige Merkmale des Problems ignoriert werden. Stimmt man den Ausführungen in Abschnitt 2 zu, so kann die Gestaltung einer computerunterstützten Lernumgebung als ein situiertes Problem aufgefasst werden. Dies deckt sich mit der Auffassung, dass Designaufgaben allgemein situierte Probleme sind (z.B. Carroll, 2000; Jonassen, 1997).

4 Vorgehensmodelle

Unter Vorgehensmodellen wird hier in Anlehnung an Floyd & Züllighoven (1997) eine Benennung und Ordnung von Aktivitäten bei der Problemlösung verstanden. Im Folgenden soll gezeigt werden, wie sich lineare, spiralförmige und zyklische Vorgehensmodelle den oben beschriebenen Problemklassen zuordnen lassen (siehe auch Lehman, 1980). Die Darstellung der Vorgehensmodelle beschränkt sich hier auf einige zentrale Merkmale. Die Unterscheidung der Vorgehensmodelle ist Floyd und Züllighoven (1997) entnommen, wobei hier eine vereinfachte Darstellungsform gewählt wurde.

4.1 Lineare Vorgehensmodelle

In linearen Vorgehensmodellen sind die einzelnen Aktivitäten sequentiell geordnet, siehe Fig. 2. Die einzelnen Phasen werden im Prinzip nur einmal durchlaufen, wobei, wenn überhaupt, nur ein Rückschritt zur vorherigen Phase vorgesehen ist. Ein solches Vorgehensmodell bietet sich für kontextunabhängige Probleme an, da die Repräsentation des Problems durch die Spezifikation gegeben und nicht veränderlich ist. Darüber hinaus führen die einzelnen Operationen zu definierten Konsequenzen, die im Hinblick auf die Zielerreichung eindeutig bewertet werden können.

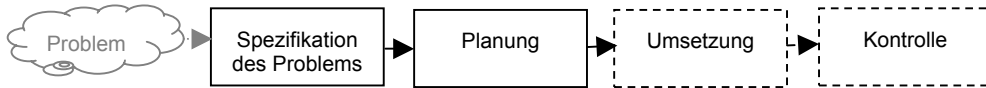


Fig. 2: Lineares Vorgehensmodell

4.2 Spiralförmige Vorgehensmodelle

In spiralförmigen Vorgehensmodellen, wie sie z.B. von Boehm (1988) entwickelt wurden, wird eine gegebene Sequenz von Aktivitäten mehrmals durchlaufen, siehe Fig. 3. An die Überprüfung in wie weit der Zielzustand erreicht wurde schließt sich in diesen Modellen eine veränderte Repräsentation des Problems an. Solche Vorgehensmodelle bieten sich für kontextabhängige Probleme an, da die gewünschte Zielerreichung oftmals erst nach mehreren Iterationen erreicht werden kann, oder aber die Umweltbedingungen, auf die sich das Problem bezieht variabel sind.

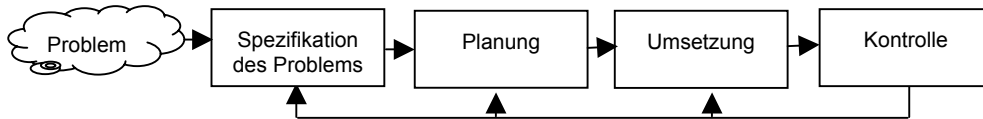


Fig. 3: Spiralförmiges Vorgehensmodell

4.3 Zyklische Vorgehensmodelle

In zyklischen Vorgehensmodellen wird eine gegebene Sequenz von Aktivitäten zunächst einmal durchlaufen. Die nächste Iteration durchläuft wieder die gesamte Sequenz, da hier davon ausgegangen wird, dass sich das Problem in Folge der Problemlösung selbst verändert hat, siehe Fig. 4. Zyklische Vorgehensmodelle bieten sich somit für situierte Probleme an.

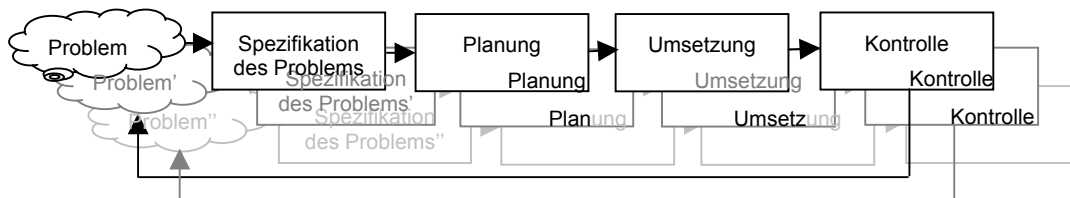


Fig. 4: Zyklische Vorgehensmodelle

5 Evaluationskonzepte

Vor dem Hintergrund der Unterscheidung von kontextunabhängigen, kontextabhängigen und situierten Problemen sowie der korrespondierenden Vorgehensmodelle können nun die Anforderungen an entsprechende Evaluationskonzepte formuliert und mit derzeitigen Evaluationsansätzen verglichen werden. Unter Evaluation wird hier der Prozess der Beurteilung eines Objektes verstanden, der dazu dient, Ungewissheit im Rahmen von Entscheidungsprozessen

zu reduzieren (siehe Mertens, 1998)². Von besonderer Bedeutung sind hier die Entscheidungsprozesse, die von den Entwicklern selbst zu treffen sind. Entscheidungsprozesse anderer Stakeholder (z.B. Geldgeber, Anwender) werden hier nicht weiter betrachtet.

5.1 Evaluation bei kontextunabhängigen Problemen

Die Evaluation bei der Lösung kontextunabhängiger Probleme ist trivial, da der Zielzustand eindeutig durch die Spezifikation definiert ist und jede Operation zu definierten Konsequenzen führt. Die Überprüfung, ob eine Lösung richtig ist, kann objektiv getroffen werden.

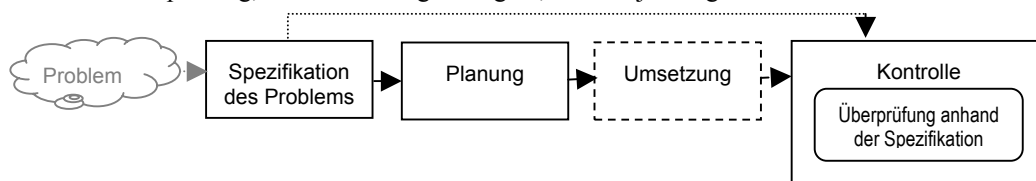


Fig. 5: Evaluation bei kontextunabhängigen Problemen

5.2 Evaluation bei kontextabhängigen Problemen

Bei kontextabhängigen Problemen setzt die Bestimmung der Qualität der Lösung einen Vergleich mit der realen Umwelt voraus. Hauptaufgabe der Evaluation beim Umgang mit kontextabhängigen Problemen ist einerseits die Bestimmung der Qualität der Lösung sowie andererseits das Auffinden von Möglichkeiten zur Verbesserung der Qualität. Da die Lösung des Problems keinen Einfluss auf das Problem hat, können zumindest theoretisch beliebig viele Problemlösungsprozesse durchgeführt werden. Die Auswahl einer „schlechten“ Lösungsstrategie ist somit im Bezug auf die reale Welt unkritisch. Unabhängig kann jedoch die Notwendigkeit das Problem zu lösen sehr hoch sein. Das Problem, Lernmaterialien zu finden, die den individuellen Anforderungen des Lernalters entsprechen verdeutlicht dies. Das Problem ist unabhängig davon, ob es Lösungsvorschläge gibt. Zu evaluieren ist, wie gut die gefundenen Lernmaterialien den Bedürfnissen des Lernalters entsprechen und ggf. wie die Ergebnisse verbessert werden können.

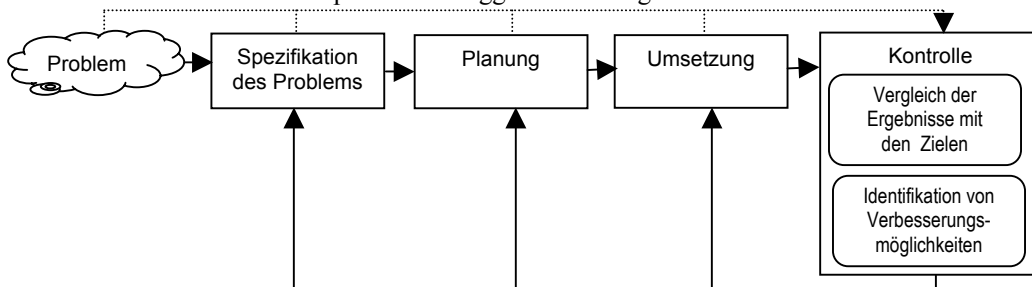


Fig. 6: Evaluation bei kontextabhängigen Problemen

² Evaluationsforschung würde zusätzlich die „explizite Verwendung wissenschaftlicher Forschungsmethoden und -techniken“ voraussetzen (Wottawa & Thierau, 1998).

Für die Evaluation kontextabhängiger Probleme eignen sich insbesondere ziel-orientierte Ansätze sowie Programmtheorie-basierte Ansätze³. Hauptziel beider Ansätze ist es zu prüfen, in welchem Ausmaß die von einem Projekt gesetzten Ziele erreicht worden sind. Programmtheorie-basierte Ansätze erleichtern darüber hinaus auch die Identifikation möglicher Schwierigkeiten bei der Umsetzung, da sie das jeweilige Programm in eine Folge von Teilschritten gliedern.

5.3 Evaluation bei situierten Problemen

Im Bezug auf situierte Probleme ergeben sich gegenüber den anderen Problemklassen erweiterte Aufgaben für die Evaluation. Da beim Umgang mit situierten Problemen stets projektunabhängige Veränderungen des Kontextes auftreten können ist es wichtig derartige Prozesse zumindest grob antizipieren zu können (vgl. Wottawa & Thierau, 1998). Da ferner jeder Problemlösungsversuch das Problem an sich verändern kann, gewinnt die Frage nach der anzuwendenden Strategie entscheidend an Bedeutung. Im Bezug auf situierte Probleme ist deshalb die Antizipation möglicher Konsequenzen einer gewählten Strategie und die Erwägung alternativer Strategien von zentraler Bedeutung (s.a. Lehman, 1980). Ferner ist im Gegensatz zu den anderen Problemklassen bei situierten Problemen das Problem an sich nicht definiert, so dass eine weitere Aufgabe darin besteht zu prüfen, ob das Problem und die Wirkungszusammenhänge in die es eingebettet ist „richtig“ verstanden worden ist (z.B. Jonassen, 1997). Erfahrungen, die innerhalb eines Zyklus gewonnen wurden, müssen im nächsten Problemlösezyklus erneut auf ihre Brauchbarkeit aufgrund des u.U. veränderten Problems überprüft werden und haben deshalb nur begrenzte Bedeutung. Aufgrund der dynamischen Eigenschaften situierter Probleme sind Repräsentationen des Problems erforderlich, die den Realitätsausschnitt abbilden können, in den das Problem eingebettet ist. Insgesamt müssen die Evaluationskriterien fortlaufend der sich wandelnden Problemstellung anpassen (vgl. Pfister & Wessner, 2000; Mayring, 2002)

Ziel-orientierte sowie Programmtheorie-basierte Evaluationsansätze weisen bezüglich situierter Probleme einige generelle Schwächen auf. So weisen z.B. Worthen et al. (1997) darauf hin, dass die Relevanz der Ziele an sich nicht hinterfragt wird, mögliche Handlungsalternativen nicht Gegenstand der Evaluation sind und nicht-intendierte Ergebnisse unberücksichtigt bleiben. Fig. 7 skizziert ein Evaluationskonzept für situierte Problemstellungen. Das Modell basiert neben den Konzepten von Lehman (1980) und Jonassen (1997) auf der Unterteilung in Kontext-, Input, Prozess- und Produktevaluation von Stufflebeam (2000).

Die Erfassung von Kontextbedingungen und das Erkennen von Entwicklungstendenzen dient dazu ein Modell des betreffenden Realitätsausschnittes zu erstellen, um eine möglichst umfassende Spezifikation des Problems vornehmen zu können. Die antizipatorische Bewertung möglicher Handlungsalternativen im Sinne einer prospektiven Evaluation erfolgt anhand des Modells, des betreffenden Realitätsausschnittes und berücksichtigt u.a. die zur Verfügung stehenden Projektressourcen wie auch mögliche Langzeit- und Nebenwirkungen. Die Bewertung der erreichten Veränderung baut auf den vorangegangenen Evaluationsschritten auf und dient dazu die Vorstellungen über das Problem und seinen Kontext zu korrigieren oder präzisieren. Insofern bereitet dieser Evaluationsschritt den nächsten Zyklus vor.

³ Programmtheorie meint in diesem Kontext die impliziten oder expliziten Überzeugungen der Projektbeteiligten warum die geplante Intervention geeignet ist die gewünschten Ziele zu erreichen (Vgl. Worthen et al., 1997)

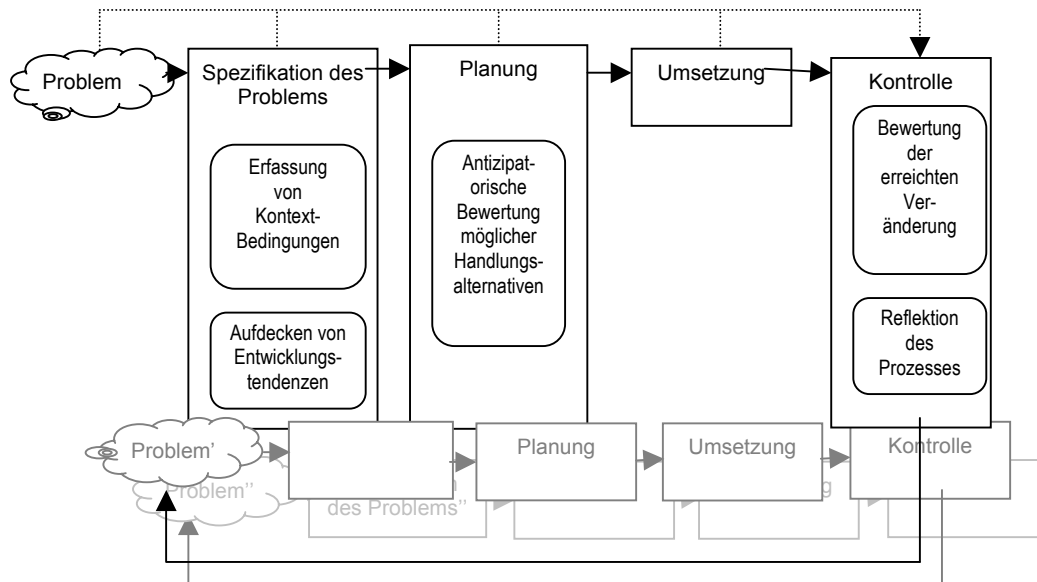


Fig. 7: Evaluation bei situierten Problemen

6 Ausblick

Aufbauend auf dem skizzierten Evaluationskonzept für situierte Probleme stehen als nächstes die Möglichkeiten zur Repräsentation sowohl komplexer Beziehungen als auch dynamischer Entwicklungen im Anwendungsfeld computerunterstützte Lernumgebungen auf unserem Arbeitsprogramm. Zur Darstellung externer Entwicklungstrends, vor deren Hintergrund die Wirkung von Handlungsalternativen erprobt werden kann, werden wir die Szenario-Technik, wie sie z.B. von van der Heyden (1996) vorgeschlagen wurde, erproben. Aus forschungsmethodischer Sicht werden wir Verfahren wie die Delphimethode, den Sociotechnical Walkthrough (Herrmann et al, 2002) oder die Claims-analysis (Carroll, 2000) erproben, mit dem Ziel, ein für Entwicklungsprojekte handhabbares Gestaltungs- und Evaluationskonzept zu entwickeln, das den Anforderungen situerter Probleme gerecht wird.

7 Literaturverzeichnis

- Boehm, B.W. (1988). A Spiral Model of Software Development and Enhancement. *Computer*. Vol. 21, Nr. 5. S. 61-72.
- Carroll, J.M. (2000). *Making use: scenario-based design of human-computer interactions*. Cambridge: MIT press.
- Dörner, D. (1993). *Die Logik des Mißlingens: Strategisches Denken in komplexen Situationen*. Reinbek: Rowohlt.

- Duncker, K. (1935). Zur Psychologie des produktiven Denkens. Berlin: Springer.
- Floyd C.; Züllighoven, H. (1997). Softwaretechnik. In: P. Rechenberg; G. Pomberger (Hrsg.). *Informatik-Handbuch*. München: Hanser Verlag.
- Herrmann, T.; Kunau, G.; Loser, K.-U. (2002). Sociotechnical Walkthrough – ein methodischer Beitrag zur Gestaltung soziotechnischer Systeme. In: Herczeg, M.; Prinz, W.; Oberquelle, H. (Hrsg): *Mensch und Computer 2002*. S. 323-332.
- Jarke, M.; Bui, X.T.; Carroll, J.M. (1998). Scenario Management: An Interdisciplinary Approach. *Requirements Engineering Journal*, Vol. 3, Nr. 3-4. S. 155-173.
- Jonassen, D.H. (1997). Instructional Design Models for Well-Structured and Ill-Structured Problem-Solving Learning Outcomes. *Educational Technology: Research and Development*. Vol. 45, Nr 1, S. 65-95.
- Keil-Slawik, R. (1999). Evaluation als evolutionäre Systemgestaltung. Aufbau und Weiterentwicklung der Paderborner DISCO (Digitale Infrastruktur für computerunterstütztes kooperatives Lernen). In: Kindt, M. (Hrsg.): *Projekterevaluation in der Lehre – Multimedia an Hochschulen zeigt Profil(e)*. Reihe: Medien in der Wissenschaft, Band 7, Münster New York München Berlin: Waxmann, S. 11–36.
- Kitchener, K.S. (1983). Cognition, metacognition, and epistemic cognition. *Human development*. Vol 26, S. 222-232.
- Lehman, M. M. (1980). Program, Life Cycles, and Laws of Software Evolution. *Proceedings of the IEEE*. Vol 68, Nr. 9, S. 1060-1076.
- Lürer, G.; Spada, H. (1992). Denken und Problemlösen. In: Spada, H. (Hrsg.). *Lehrbuch allgemeine Psychologie*. 2. korrigierte Aufl.- Bern: Huber. S. 189-280.
- Mayring, P. (2002). *Einführung in die qualitative Sozialforschung – Eine Anleitung zu qualitativem Denken*. 5. überarb. Aufl. – Weinheim: Beltz.
- Pfister, H.-R.; Wessner, M. (2000). Evaluation von CSCL-Umgebungen. In H. Krahn; J. Wedekind (Hrsg.). *Virtueller Campus, 99. Heute Experiment – morgen Alltag? Medien in der Wissenschaft*. Band 9. Münster: Waxmann, S. 139-149.
- Stufflebeam, D.L. (2000). The CIPP Model for Evaluation. In: D.L. Stufflebeam; G.F. Madaus; T. Kellaghan (eds.). *Evaluation Models: Viewpoints on Educational and Human Services Evaluation*. 2nd ed. Boston: Kluwer, S. 279-318.
- Van der Heyden, K. (1996). *Scenarios: The Art of Strategic Conversation*. Chichester: Wiley & Sons.
- Worthen, B.R.; Sanders, J.R.; Fitzpatrick, J.L. (1997). *Program Evaluation: Alternative Approaches and Practical Guidelines*. 2nd ed. New York: Addison Wesley.
- Wottawa, H.; Thierau, H. (1998). *Lehrbuch Evaluation*. 2. vollst. ueberarb. Aufl.- Bern: Huber.
- Zimbaro, P.G. (1995). *Psychologie*. 6. neu bearb. u. erw. Aufl.- Berlin: Springer.

Kontaktinformationen

Christoph Richter
 Learning Lab Lower Saxony, Universität Hannover
 Expo Plaza 1, 30539 Hannover
 Email: richter@learninglab.de
 Tel.: 0511 762 9754