

# Anforderungsanalyse für ein Werkzeug zur Gestaltung von webbasierten Studieneinheiten für den Studiengang Klassische Altertumswissenschaften

Alexander Fronk  
Universität Dortmund  
Lehrstuhl Software-Technologie

Ulrich-Walter Gans  
Ruhr-Universität Bochum  
Institut für Archäologie

14. Dezember 2001

## Zusammenfassung

Die Anforderungsanalyse in Software-Projekten erfordert stets ein hohes Maß an Kommunikation zwischen Kunden und Entwicklern: Eigenschaften des zu erstellenden Software-Produkts müssen erfaßt, Anforderungen an das System und Erwartungen der Kunden umgesetzt werden. Die UML bietet hier mit ihren **Use Cases** lediglich eine nicht kundenspezifische und semi-formale Herangehensweise; formale Methoden sind nicht bekannt. In Abhängigkeit der informations-technischen Vorbildung des Kunden erweist sich nicht nur das Erfassen von Anforderungen und Wünschen als problematisch; viel schwieriger gestaltet sich die Diskussion ihrer Umsetzung. Informatiker müssen hier nicht zuletzt wegen fehlender Kommunikationsmethoden allgemein verständlich und dennoch technisch zielgenau argumentieren.

Dieser Artikel ist bestrebt, das *inkrementelle Prototyping* als eine softwaretechnische Herangehensweise im Hinblick auf das Erfassen und Diskutieren von Anforderungen, die Wissenschaftler aus geisteswissenschaftlichen Disziplinen an eine Lehr/Lern-Software stellen, zu motivieren und dieses Entwicklungsparadigma damit dem *Requirements-Engineering für Geisteswissenschaftliche Anwendungen* zugänglich zu machen. Eine geeignete Adaption hier anzuwenden wird nicht zuletzt durch die breite Einsatzmöglichkeit des Prototyping untermauert (vgl. etwa [ZAD93]).

## 1 Das Projekt Pharos

Im Rahmen des Projektes *Pharos* soll ein Werkzeug zur Gestaltung von webbasierten Studieneinheiten für den Studiengang *Klassische Altertumswissenschaften* im Rahmen von Bachelor- und Master-Abschlüssen an den Universitäten Bochum und Köln entstehen mit dem Ziel, Studierenden möglichst früh-

zeitig fachliche Orientierungshilfen anzubieten und die starke Interdisziplinarität des Studienganges durch eine Annäherung an die Nachbarfächer und durch Adaptionen aus deren methodischem Apparat deutlich zu machen.

Das Projekt wird als Kooperation durchgeführt. Beteiligt sind der Lehrstuhl für Software-Technologie der Universität Dortmund, das Institut für Archäologie der Ruhr-Universität Bochum sowie die Institute *Klassische Archäologie*, *Alte Geschichte* und *Klassische Philologie* der Universität zu Köln.

Die Studieneinheiten umfassen Lernmodule zu den Themenkomplexen *Homerisches Griechenland*, *Klassisches Athen*, *Augusteisches Rom* und *Das Adoptivkaisertum*. Jeder der drei Disziplinen *Klassische Archäologie*, *Alte Geschichte* und *Altphilologie* obliegt dabei die Aufgabe, sowohl jeden Themenkomplex aus ihrer Sicht zu beleuchten, die inhaltlichen Bezüge zu den Nachbarfächern aufzuzeigen als auch rechnergestützt unter Verwendung hypermedial verknüpfter Texte, Grafiken, Fotos etc. zu präsentieren. Die zunächst von den jeweils anderen Disziplinen losgelösten Beiträge sollen durch Ausnutzung hypermedialer Verknüpfungen nicht nur zu einer Interaktivität mit den Lernenden führen, sondern insbesondere inhaltliche Querbezüge zwischen den einzelnen Disziplinen explizit machen. Das Erstellen und Ändern der Inhalte soll ebenfalls webbasiert durch dasselbe Werkzeug unterstützt werden. Dadurch werden die Lernmodule nicht auf eine zuvor fest definierte Anzahl von Beiträgen ausgerichtet und somit kein inhaltlich unveränderbares Lernmedium realisiert.

Informatiker stehen hier vor der schwierigen Aufgabe, die Anforderungen der Geisteswissenschaftler an eine solche Software mit diesen gemeinsam zu erarbeiten. Die gefundenen Anforderungen müssen dann umgesetzt und die Umsetzung wiederum diskutiert und validiert werden. Es ist also insbesondere der Fra-

ge nachzugehen, welche Methoden und Prinzipien der Software-Technik eingesetzt werden können, um mit technisch unvorbelasteten „Kunden“ über das Produkt kommunizieren zu können.

## 2 Inkrementelles Prototyping zur Anforderungsanalyse

In einer ersten Annäherung an das Softwareprodukt ist seine Zweiteilung festzustellen: Aus Sicht der Lernenden sind Inhalte interaktiv anzuschauen; aus Sicht der Lehrenden sind eben diese Inhalte zu erstellen und hypermedial zu verknüpfen. Ebenso ist es nötig, ein gemeinsames Lehrziel zu arbeiten, das alle Lernmodule umspannt.

Zur Erfassung dieser ersten Anforderungen und möglichen Verfeinerungen helfen die aus der UML bekannten Use Cases [FS00]. Die Projektpraxis jedoch zeigt, daß die Diskussion der Anforderungen dann bereits nicht mehr verständlich durchgeführt werden kann: Die Erklärung der UML-Notation gelingt noch recht einfach, aber es fehlt das pragmatische Verständnis dessen, was dort beschrieben wird. Die Frage nach einer „begreifbaren“ Umsetzung der Anforderungen tritt hier sehr schnell in den Vordergrund.

Eine Lösung dieses nicht unbekanntes Phänomens bietet das *inkrementelle Prototyping* [Som01] an: es gelingt, einen Prototypen zu bauen, der Anforderungen umsetzt und mit dessen Hilfe diskutiert werden kann, ob 1.) die richtigen Anforderungen erfaßt wurden und ob 2.) die Anforderungen korrekt, und das heißt im Sinne des Kunden, umgesetzt wurden. Diese Fragestellung treibt die Forderung nach Validierung und formaler Verifikation an [GB99]. Unserer Erfahrung nach ist die Formalisierung der Anforderungen jedoch erst ein zweiter Schritt, der hilft, weitere Wünsche auf Erfüllbarkeit oder Widersprüchlichkeit zu untersuchen vor allem dann, wenn die Zahl der Anforderungen und ihre Komplexität ein individuell zu ermessendes Maß an Verständlichkeit überschreitet oder es notwendig ist, die beweisbare Korrektheit einer Implementierung zu verlangen. Formalisierung hilft also, die Erfüllbarkeit von Anforderungen zu beweisen, trägt aber in unserem Fall nicht zu deren anschaulicher Darstellung bei.

Eben dieser Anschauungsaspekt treibt uns an: Die im inkrementellen Prototyping möglichst früh angesiedelte Entwicklung eines lauffähigen Produkts macht die Umsetzung von Anforderungen gerade dadurch visuell begreifbar, daß der Prototyp als Modell seiner selbst dient. Die Umsetzung von Anforderungen kann also am Prototypen studiert, bestätigt oder verworfen werden, Änderungswünsche können konkret aufgezeigt werden.

## 3 Auswirkungen des Anschauungsaspekts auf den Projektverlauf

In einer ersten Verfeinerung der Anforderungen wurde festgelegt, die Interaktion der Studierenden mit dem Produkt zunächst in den Hintergrund zu stellen und die Inhalte selbst zu betrachten: Sie zu erstellen und einzusetzen stellt höhere Anforderungen.

Um festzustellen, wie die Wissenschaftler/innen der einzelnen Disziplinen jeweils an die Erstellung hypermedialer Präsentationen herangehen möchten, wurden zunächst konzeptionelle Gemeinsamkeiten erkundet und die Ergebnisse grafisch fixiert. Hieran konnten Wünsche und Vorstellungen diskutiert werden, die dann zu einem ersten technischen Entwurf, der Architektur des Werkzeugs, führten.

Es offenbarte sich im Projekt *Pharos* ein dreistufiges Vorgehen bei der Erstellung der Lernmodule, das als Prozeßmodell verstanden werden kann: von der Auswahl eines Seitenlayouts über die Erstellung der Inhalte hin zu deren hypermedialen Verknüpfung. Die Modellierung dieses Prozesses ist für Informatiker interessant, da sowohl Schnittstellen zu Datenbanken zwecks Speicherung der Inhalte als auch die zur Unterstützung dieses Vorgehens nötigen Funktionalitäten des Werkzeugs spezifiziert und implementiert werden müssen; für die anderen Projektpartner muß auch hier wieder auf Transparenz der Ergebnisse Wert gelegt werden: der Prototyp wird nun so erweitert, daß die Erstellung von Lernmodulen nachvollziehbar wird und praktisch erprobt werden kann. Diese Experimentierphase dient der Validierung des Prozeßmodells.

Die Auswahl konkreter Modulinhalt, die bei den Geisteswissenschaftlern nicht nur eine hohe Priorität sondern auch das größte Interesse besitzt, kann weiter voran getrieben und der Prototyp dabei als Werkzeug gleichsam benutzt und validiert werden. Somit wird die Auswahl der Inhalte explizit in den Prototyping-Ansatz eingebunden und die für die geisteswissenschaftlichen Disziplinen notwendigen Schritte mit den software-technischen synchronisiert.

Diesen Prozeß gilt es im Laufe des Projekts weiter zu verfolgen und umzusetzen. Dazu wird der Prototyp schrittweise derart erweitert, daß die Umsetzung von Anforderungen anschaulich bleibt und damit deren Diskutierbarkeit in den einzelnen Entwicklungsphasen stets gewahrt wird.

## 4 Bezug zu Hypermedia Entwicklungsprozessen

Ein Vorgehen zur Erstellung hypermedialer Präsentationen schlägt zum Beispiel das Prozeßmodell OOHDM [SR95] vor. Das Modell ist sicher geeignet, technisch versierte Anwender zu unterstützen: Geisteswissenschaftler hingegen bringen die nötigen Voraussetzungen zum Verständnis dieser Methode nur bedingt mit. Dennoch sind solche Prozeßmodelle auch in Pharos einsetzbar: es muß wieder eine geeignete Anschauung der Ergebnisse erarbeitet werden.

Das Erstellen und insbesondere das Verlinken von Texten und Grafiken stellt eine weitere Hürde dar, die es zu nehmen gilt. Wie oben diskutiert sollen die Inhalte der Lernmodule nicht nur von den Geisteswissenschaftlern selber erstellt, sondern auch von ihnen hypermedial verknüpft werden. Erfahrungen aus dem *Altenberger Dom-Projekt* [ADK01] zeigen, daß eine selbst in Absprache mit den Anwendern konzipierte XML-Sprache die Erstellung und Verlinkung von Daten zwar massiv erleichtert, aber dennoch sehr schnell zu komplex wird, um die nötige Akzeptanz zu finden.

Auch hier muß über eine geeignete Anschaubarkeit nachgedacht werden. So kann eine grafische Schnittstelle zu einer XML-Sprache, die auf die Bedürfnisse der Texterstellung und Verlinkung in Pharos ausgerichtet ist, Abhilfe schaffen: Links müssen dann nicht mehr durch *Tags* von Hand innerhalb des Textes geschrieben, sondern können per Knopfdruck in den Text eingefügt werden; eine entsprechende Funktionalität ermöglicht das Verlinken von ausgezeichneten Bereichen in Grafiken.

Andere Entwicklungsparadigmen wären analog zu behandeln: es sollte stets auf die Umsetzung des Prozesses und die Transparenz der erzielten Ergebnisse geachtet werden.

## 5 Fazit

Der bisherige Projektverlauf gibt Anlaß zu der Hoffnung, daß 1.) die gewählte Form der Anforderungsanalyse ein technisch qualitativ hochwertiges und gleichsam akzeptiertes Produkt zu erstellen hilft und 2.) die gezeigte Adaption des inkrementellen Prototyping im Sinne von Lowe und Hall der Definition des *Hypermedia Engineering* [LH99] genügt. Zu beobachten ist, daß dieser aus einem allgemeinen softwaretechnischen Vorgehen abgeleitete Prozeß speziell auf hypermediale Anwendungen abzielt und gleichzeitig auf Geisteswissenschaftler ausgerichtet wird. Es zeigt sich wieder einmal, daß die „traditionelle Software-Technik“ hinreichend starke Methoden und Prinzipien

besitzt, die mächtig genug sind, um auch die Anforderungsanalyse für Geisteswissenschaftliche Anwendungen greifbar und gestaltbar zu machen. Der Zugang zu den Ergebnissen der Adaption ist hingegen stark durch die Anwender geprägt.

**Danksagung** Wir danken Alexander Papaspyrou und Martin Testrot für die technische Unterstützung bei der Realisierung des Prototypen.

## Literatur

- [ADK01] ALFERT, K., E.-E. DOBERKAT und C. KOPKA: *Towards Constructing a Flexible Multimedia Environment for Teaching the History of Art*. Memorandum 101, Lehrstuhl für Software-Technologie, Universität Dortmund, September 2001. ISSN 0933-7725.
- [FS00] FOWLER, M. und K. SCOTT: *UML konzentriert*. Addison-Wesley, 2000.
- [GB99] GAUDEL, M.-C. und G. BERNOT: *The Role of Formal Specifications*. In: ASTESIANO, E., H.-J. KREOWSKI und B. KRIEG-BRÜCKNER (Herausgeber): *Algebraic Foundations of Systems Specification*, IFIP State-of-the-Art Report, Seiten 1 – 12. Springer, 1999.
- [LH99] LOWE, D. und W. HALL: *Hypermedia & the Web - an engineering approach*. Wiley & Sons, 1999.
- [Som01] SOMMERVILLE, IAN: *Software Engineering*. Addison-Wesley, 2001. Sechste Edition.
- [SR95] SCHWABE, D. und G. ROSSI: *The Object-Oriented Hypermedia Design Model*. Communications of the ACM, 38(8):45–46, August 1995.
- [ZAD93] ZÜLLIGHOVEN, H., W. ALTMANN und E.-E. DOBERKAT (Herausgeber): *Requirements Engineering '93: Prototyping*, Band 41 der Reihe *Berichte des German Chapter of the ACM*. Teubner, 1993.