

Aspekte der Softwareentwicklung im Team unter Berücksichtigung gruppenpsychologischer Phänomene

Management von Software-Projekten, Sommersemester 2001

Gerald Wagenknecht
Matr.-Nr.: 9700770
wagen@informatik.tu-cottbus.de

Inhalt

Inhalt	2
1 Einführung	2
2 Probleme der Softwareentwicklung	3
3 STEPS	4
4 Der dialogische Softwareentwurf.....	12
5 Umsetzung im Softwarepraktikum	16
Literatur	17
Anhang A: Projektstagebuch der Gruppe: Formeditor.....	17
Anhang B: Projektstagebuch der Gruppe: Khepera	23
Anhang C: Was habe ich aus dem Softwarepraktikum gelernt ?	28

1 Einführung

Anhand des Buches "Software-Entwicklung im Team" von Jürgen Pasch werden Probleme der Gruppenarbeit bei Softwareprojekten erläutert. Dazu wird auch auf gruppenpsychologische Aspekte (Manfred Sader: "Psychologie der Gruppe) eingegangen. Mit der von Pasch propagierten prozeßorientierten Sichtweise der Software-Entwicklung und dem dialogischen Software-Entwurf werden Lösungsansätze beschrieben.

Kapitel 2 listet typische Probleme auf, die bei der Entwicklung von Software auftreten. Der theoretische und methodische Ansatz STEPS (Software-Technik für Evolutionäre Partizipative Systementwicklung) wird in Kapitel 3 vorgestellt. Darauf aufbauend beschreibt Kapitel 4 den dialogischen Softwareentwurf. Zu guter Letzt wird die praktische Umsetzbarkeit für das Softwarepraktikum geprüft (Kapitel 5).

2 Probleme der Softwareentwicklung

Softwareprodukte sind schlecht! Sie werden zu spät an die Kunden ausgeliefert! Sie sind fehlerhaft und entsprechen nicht den Anforderungen! Die Kunden sind unzufrieden! Die "Reparatur" fehlerhafter Software verschlingt Millionen von Mark! Warum ist das so? Warum stecken wir immer noch in der schon vor 30 Jahren propagierten Softwarekrise? Die Ursachen dafür sind vielfältig.

Software wird immer größer. Unter dem Druck der enormen Expansion der DV-Industrie wird versucht, die Probleme durch den Einsatz von Hunderten und Tausenden von Programmierern wettzumachen. Gegen Ende der 60er Jahre hat diese Entwicklung ihren Höhepunkt erreicht und geht als die Softwarekrise in die Geschichte ein.

Software wird immer komplexer. Dieser Komplexität wird eine komplexe Bürokratie entgegengestellt. Eingebettet in eine formale Organisationsstruktur werden dem einzelnen Entwickler zahlreiche Aktivitäten und Ergebnisse abverlangt, die mit einer Softwareentwicklungsumgebung operationalisiert und verwaltet werden.

Ein weitere Ursache liegt in der Organisationsstruktur und den Einfluß von Macht und Hierarchie auf die Projektarbeit. Ein Projektleiter hat zwar die Verantwortung für ein Projekt, aber keine Kompetenzen. Beklagt wird die Ausstattung an Weisungs- und Entscheidungskompetenzen der Projektleitung für das eigene Projekt, die sich aus der organisatorischen Einbettung des Managements in die funktionale Stammorganisation des Unternehmens ergibt. Mittlerweile halten viele Organisationstheoretiker hierarchische Organisationsformen zur Bewältigung von komplexen Arbeits- und Entscheidungsprozessen für gänzlich ungeeignet.

Als Zauberformel für den Ausweg aus der Softwarekrise wurde eine Linearisierung und Mechanisierung des Entwurfs- und Testprozesses propagiert, also Entwerfen, Testen, Programmieren, Organisieren. Einige "Software-Gurus" der damaligen Zeit sahen die Probleme bei den beteiligten Softwareentwicklern. Schwartz formulierte acht Kernprobleme (vgl. [Pa94], S.34/35):

- Keiner hat ein Gesamtverständnis der Problemstellung.
- Die Programmierer kommunizieren schlecht miteinander.

- Innerhalb einer Gruppe von Programmierern gibt es erhebliche Unterschiede in ihren Fähigkeiten.
- Gute Programmierer sind kreativ. Sie sind aber auch in Situationen kreativ, die gar keine Kreativität erfordern.
- Nicht-Programmierer verstehen die Programmierer nicht. Bekommen diese mehr Verantwortung, hält niemand sie davon ab, immer mehr Programmierer anzustellen.
- Es ist sehr schwierig, von Programmierern den Projektfortschritt zu erfahren. Zu dem sind die meisten Manager sehr leichtgläubig. Die Methoden zur Vorhersage des Projektfortschrittes sind unglaublich schlecht.
- Die Tester eines Systems haben keine Methoden, die gewährleisten, daß das System umfassend getestet ist.
- Kein Mensch kennt das Gesamtsystem oder seinen augenblicklichen Zustand.

Außerdem bemerkt er noch: "Ich muß zugeben, ich habe mich dauernd gefragt, ob diese Systeme so viele Leute benötigen, weil sie groß sind, oder ob diese Systeme so groß sind, weil so viele Leute an ihnen arbeiten..."

3 STEPS

Die produktorientierte Sichtweise war um 1970 eine wichtige Errungenschaft als Antwort auf die Software-Krise. Sie betrachtet Software als eigenständiges Produkt aus Programmen und Dokumentation und die Softwareentwicklung als Produktion im Sinne einer Fertigung. Maßgebliche Ziele sind dabei die Kontrollierbarkeit des Herstellungsprozesses, die Gewährleistung vorgegebener Anforderungen und Qualitätsmerkmale durch die Produkte und die Einhaltung eines fest kalkulierbaren finanziellen und terminlichen Rahmens. Ein typisches Modell dieser Sichtweise ist das Wasserfallmodell. (vgl. [Pa94], S.57)

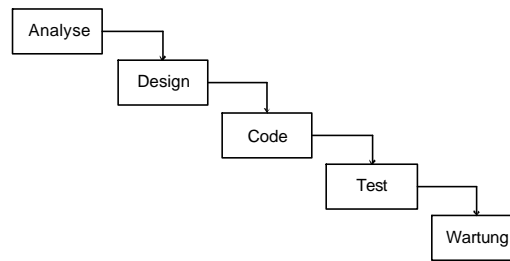


Abbildung 1: Wasserfallmodell als typisches Modell der Produktsicht

Phasenmodelle liefern statische Beschreibungen der Softwareentwicklung vor oder nach der jeweiligen Projektsituation. Aber die Softwareentwicklung in Projekten ist nicht alleine eine mathematische, technologische oder administrative Herausforderung, sondern ein komplexer *sozialer Gruppenprozeß*, in dem die dynamisch ablaufenden Verständigungsprozesse maßgeblich die Qualität des Produktes beeinflussen. Daraus zog Pasch Schlußfolgerungen für die Projektgestaltung und den Softwareentwurf. Diese werden in diesem und dem nächsten Kapitel vorgestellt.

STEPS (Software-Technik für Evolutionäre Partizipative Systementwicklung) ist ein theoretischer und methodischer Ansatz der Software-Technik, welcher von Christiane Floyd seit Beginn der 80er Jahre an der TU Berlin entwickelt wird. Die wichtigsten Bestandteile sind (vgl. [Pa94], S. 56):

- eine prozeßorientierte Sichtweise der Software-Entwicklung, die als grundlegendes Paradigma für die Software-Technik, anstelle der herkömmlichen, produktorientierten Sichtweise gesetzt wurde;
- ein menschenzentrierte Qualitätsbegriff, der sich vorrangig am angestrebten Einsatz von Software, und erst an zweiter Stelle an Eigenschaften der Software selbst orientiert;
- ein zyklisches Projektmodell, das die Software-Entwicklung in Zyklen anordnet und als Rahmen angesehen wird, der in einer konkreten Situation mit der jeweils relevanten Strategie auszufüllen ist;
- Methodenkomponenten für die Schwerpunkte Anforderungsermittlung, Dialogschnittstellenentwicklung, Softwareentwurf und zur systematischen Programmierung;

- Projekttechniken, welche kreative Zusammenarbeit in Gruppen und die Kommunikation zwischen Entwicklern und Benutzern fördern.

Bei Floyd/Pasch wird die Softwareentwicklung als eine spezifische Ausprägung von Design gesehen. Unter Design wird der kreative Vorgang verstanden, indem das Problem erschlossen, eine zugehörige Lösung erarbeitet und in menschlichen Sinnzusammenhängen gepaßt wird. Dieser Design-Begriff geht auf Naur zurück. Softwareentwicklung wird ganzheitlich als Zusammenspiel von Entwurf und Gestaltung gesehen. Design betrifft sowohl das Produkt Software als auch seine Einpassung in den Einsatzkontext und den Entwicklungsprozeß selbst. Man muß also immer die Gestaltung von Technik und Arbeit zugleich im Auge haben. Die Arbeitsinhalte, die Arbeitstätigkeiten und die Arbeitsorganisation bestimmen die Qualitätsmerkmale des Softwaresystems. Umgekehrt wirkt das entwickelte Softwaresystem auf die Arbeitsprozesse und verändert sie. (vgl. [Pa94], S.60)

Ein auf die Interessen der Benutzer ausgerichtete Entwicklung von Software kann weder von den Entwicklern noch von den Benutzern allein durchgeführt werden. Die Entwickler sind die Experten im Bereich der Computertechnologie und die Benutzer verfügen über Fachwissen im Einsatzbereich der Software. Pasch sieht die *gleichberechtigte Partizipation*, (also Beteiligung) der Benutzer und Entwickler am Entwicklungsprozeß als eine notwendige Bedingung für die Herstellung einer gemeinsamen Gestaltungskompetenz. Weiterhin ist eine vollständige Ermittlung der Anforderungen hinsichtlich des Funktionsumfangs und der Qualitätsmerkmale des Softwaresystems prinzipiell unmöglich. Eine auf Sozialverträglichkeit und Wirtschaftlichkeit abziehende Systementwicklung muß von der evolutionären Entwicklung des Umfeldes der eingesetzten Software ausgehen. Es verändern sich beispielsweise normative, ökonomische und technologische Gegebenheiten, die unter Umständen auch durch den Einsatz der Software hervorgerufen werden können. Hieraus erwachsen veränderte und neue Anforderungen an das eingesetzte System. (vgl. [Pa94], S.61)

Die oben genannten Aspekte zur Softwareentwicklung sind ausschlaggebend für das zyklische Projektmodell. Die Systementwicklung wird hier als Folge von Entwicklungszyklen angesehen. Jeder Zyklus hat die Herstellung und den Einsatz einer Systemversion zum Gegenstand. Das Versionskonzept berücksichtigt die evolutionäre Veränderung der Anforderungen der Benutzer. (vgl. [Pa94], S.62)

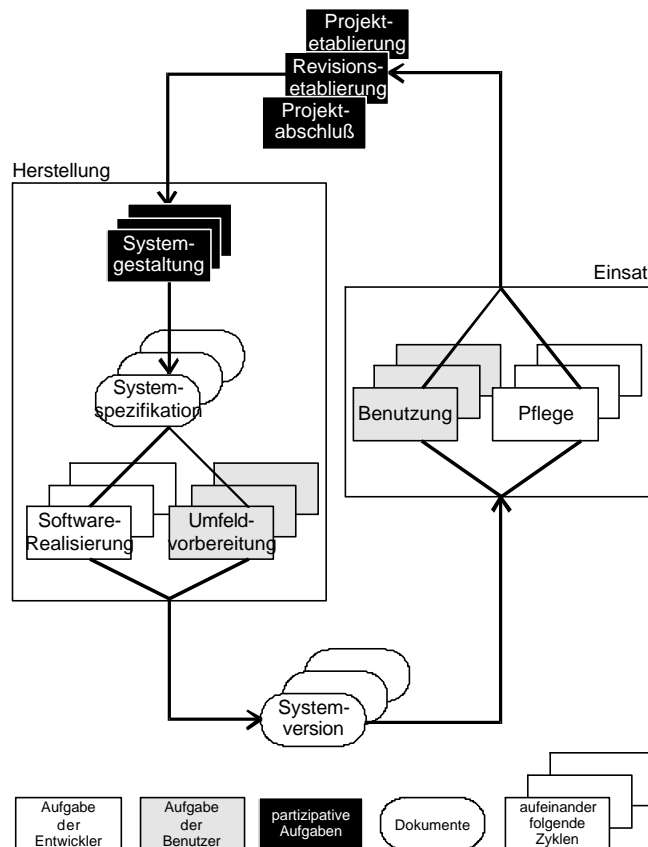


Abbildung 2: Das Projektmodell von STEPS (aus [Pa94], S.63)

Die sich aus dem Einsatz der Software ergebenden Veränderungen der Anforderungen müssen von dem Projektmodell systematisch mit einbezogen und methodisch umgesetzt werden. Ein auf Systemversionen bezogenes Projektmodell erlaubt auch die methodische Planung von Revisionen gleich zu Beginn eines Entwicklungszyklus. Sind die zu lösenden Probleme sehr komplex und die Anforderungen sehr umfangreich, so kann die Planung mehrerer Versionen als methodisches Mittel zur Anwendung kommen, um den Entwicklungsprozeß durchschaubarer zu gestalten. (vgl. [Pa94], S.62)

Das Projektmodell reflektiert die Sicht der Entwickler ebenso wie die Sicht der Benutzer und weist die gemeinsamen und gesonderten Arbeitsprozesse beider Gruppen aus. Alle am Projekt Beteiligten müssen bereit sein, sich auf kooperative Arbeitsprozesse zur Entwicklung des Systems einzulassen und sich auf eine gemeinsame Vorgehensweise und die hiermit verbundenen Verpflichtungen zu einigen. Ein allen Beteiligten gemeinsames Verständnis vom Projekt und seinen Zielen ist die zentrale Voraussetzung für die Durchführung der Systementwicklung. (vgl. [Pa94], S.62/63)

Das Ergebnis der *Projektetablierung* ist ein grobes *Systemkonzept*, aus dem sich die produktorientierten Aktivitäten ergeben, sowie ein *Projektplan*, aus dem sich die

prozeßorientierten Aktivitäten herleiten. Dabei diskutiert die Projektgruppe folgende Problempunkte (vgl. [Pa94], S.69):

- Ist die Beschreibung des Auftrags klar formuliert? Welche Verpflichtungen gehen beide Seiten ein?
- Wie werden die verschiedenen beteiligten Interessengruppen Einfluß auf das Projekt nehmen?
- Muß noch eine Durchführbarkeitsstudie und Anforderungsanalyse durchgeführt werden?
- Welche Rolle werden die Benutzer spielen?
- Ist das Projekt an Produktmuster und Standards gebunden?
- Welche generellen Ausstattung steht zur Verfügung? Kann das Projekt die nötigen Ressourcen erhalten? Werden diese noch von anderen Projekten genutzt?

Zur Etablierung der Projektgruppe klären die Mitglieder Fragen zur Kooperation, Projektgestaltung und Wissensakquisition. Für die Kooperation muß sie Regeln zur Zusammenarbeit festlegen (vgl. [Pa94], S.70):

- Wie soll sich die Gruppe verhalten, wenn sie Ideen produziert?
- Wie soll sich die Gruppe verhalten, wenn sie nach verschiedenen Vorschlägen arbeitet?
- Wie werden Entscheidungen in der Gruppe getroffen? Entscheidet der Projektleiter allein, gibt es Gruppenkonsens, oder ein Recht auf Veto?
- Welches Verhalten wird akzeptiert und welches nicht? Wie verhält sich die Gruppe bei persönlichen Konflikten?
- Wie werden neue Regeln eingeführt?

In diesem Punkt lohnt es sich, einen allgemeinen Blick in psychologische Betrachtungen der Gruppenforschung zu werfen und die Begriffe der *Gruppe*, *Gruppenziel*, *Konformität*

und *Rollenkonzept* und zu klären. Die Gruppenforschung beschäftigt sich unter anderem mit folgenden Fragestellungen (vgl. [Sa94], S.20):

- Ist die Risikobereitschaft des Menschen größer bei alleiniger Verantwortung des Einzelnen oder bei gemeinsamer Verantwortung von Gruppen?
- Arbeitet eine Gruppe besser bei hohem oder bei geringem inneren Zusammenhalt dieser Gruppe?
- Wie kann man Informationsaustausch und die Interaktion in Gruppen verbessern?
- Wie kann man Entscheidungsprozesse in Gruppen optimieren?
- Wie groß darf eine Gruppe sein, wenn eine Entscheidung von allen gemeinsam getragen werden soll?
- Was sind wesentliche Eigenschaften, die ein Führer einer Gruppe haben sollte?
- Wie kann ein Führer einer Gruppe diese für vorgegebene Ziele motivieren?

Zunächst sollte der Begriff "Gruppe" definiert werden. Es gibt aber leider keine allgemeingültige Definition in diesen Bereich. Der Begriff der "Gruppe" ist eher ein Konstruktbegriff. Hier nun drei verschiedene Definitionen (vgl. [Sa94], S.37/38):

- "Wenn zwei oder mehr Personen in irgendeiner Beziehung zueinander stehen, so bilden sie eine Gruppe." (Lindgren in einer Einführung in die Sozialpsychologie).
- "Eine Gruppe kann definiert werden als eine Mehrheit von Individuen, die in Kontakt miteinander stehen, aufeinander reagieren und in wesentlichen Punkten Gemeinsamkeiten erleben." (Olmsted in einer Einführung in die Kleingruppenforschung).
- "Eine sozialpsychologische Gruppe ist ein organisiertes System von zwei oder mehr Individuen, die so miteinander verbunden sind, daß in einem gewissen Grade gemeinsame Funktionen möglich sind, Rollenbeziehungen zwischen den Mitgliedern bestehen und Normen existieren, die das Verhalten der Gruppe und aller ihrer Mitglieder regeln." (McDavid & Harari, ebenfalls in einem Lehrbuchtext).

Es ist ebenso schwierig den Begriff des "Gruppenzieles" zu definieren. Mackie und Goethal schlagen folgende Klassifikation von Gruppenzielen vor. Da gibt es zunächst *praktische Ziele* wie Problemlösen, Leiten einer Organisation oder andere praktische Aufgaben, die ein Handeln in einer Gruppe erfordern. Dann gibt es die große Klasse der Gruppen, die ihren Mitgliedern *Kenntnisse und Informationen* verschaffen. Die Skala reicht vom Schützenverein über den Kegelclub bis hin zu offiziellen Berufsorganisationen. Hierbei sind die vorgeblichen Vereinsziele häufig völlig belanglos. Die dritte Klasse sind Gruppen, die ihren Mitgliedern *soziale Identität* verschaffen. Als Musterbeispiel dienen die englischen Clubs oder hierzulande die Rotary Clubs. (vgl. [Sa94], S.65)

Wenn bei der Bearbeitung von Gruppenproblemen von "Konformität" die Rede ist, so ist damit im allgemeinen die Beeinflussung des Denkens und/oder Verhaltens von Gruppenmitgliedern in Richtung auf Gruppenregeln oder Gruppen-Normen gemeint: Gruppenmitglieder erleben oder unterstellen stillschweigend und oft unthematisch,

- daß die Gruppe etwas von ihnen erwartet;
- daß ein bestimmtes Verhalten zulässig oder unzulässig ist;
- daß von der Gruppe ein Druck auf Handeln, Verhalten, Unterlassen ausgeübt wird.

In den weitaus meisten Fällen steht das nicht irgendwo geschrieben, sondern "ist einfach so". Es handelt sich um subjektive Phänomene, um erlebte Gebote, Verbote oder Üblichkeiten. Als Ausgangspunkt oder "Sender" wird manchmal explizit die Gruppe gesehen, vielfach sind solche erlebten Forderungen und Regeln aber einfach da. Das statisch gedachte Endergebnis der Übereinstimmung oder Einheitlichkeit und die erlebte Kraft als Druck auf die Einhaltung wird als Konformität bezeichnet.

Der Begriff der "Rolle" wird in der Literatur vielfältig und sehr unterschiedlich verwendet. Auch ganz allgemein zur Beschreibung von Rollenstrukturen ist es oft hilfreich Rollenbegriffe zu benutzen. Sader schreibt: "Als *aufgabenorientierte* Positionen wären etwa der Koordinator, der kritische Bewerter, der Verfahrens- und Durchführungstechnologe zu nennen; an *gruppenprozeßorientierten* Positionen wären etwa die des Ermutigers, des Konfliktharmonisierers, des Kommentators der Gruppenprozesse oder des vorbildlichen Gruppenmitgliedes zu nennen; an *individuumszentrierten* Positionen nennen Benne & Sheats unter anderem den Aggressiven, den Quertreiber, den Von-sich-selbst-Redner, den Dominierenden oder den Unterstützungsbedürftigen.". Solche Bezeichnungen für das Verhalten der Gruppe mögen nützlich sein und zur Verständigung beitragen, aber ein dogmatisches Festhalten an einem solchen Rollenkonzept erscheint

unsinnig, da Rollen situativ bedingt wechseln können und der Beobachter der Gruppe ordnet ("preßt") die Beteiligten oft einer Rolle zu (A ist ein Hemmschuh für jede sachliche Arbeit).

Nachdem Exkurs in die Gruppenpsychologie kommen wir wieder zurück zum zyklischen Projektmodell. Der erste *Entwicklungszyklus* wird durch die von den Entwicklern und Benutzern gemeinsam vollzogene Projektabklärung initiiert. Sie dient zur kooperativen Erarbeitung des Projektplans. Jeder *Revisionszyklus* wird durch eine *Revisionsetabklärung* initiiert. Sie wird auf der Grundlage der sich aus dem Einsatz der vorherigen Version ergebenden Anforderungen durchgeführt. Auch diese wird durch Entwickler und Benutzer kooperativ vereinbart. (vgl. [Pa94], S.64)

Bei der *Systemgestaltung* geht es zunächst um die Ermittlung der Anforderungen und um die Gestaltung der Benutzung. Die *Systembenutzung* beinhaltet die Einbettung des Rechners in die Arbeitstätigkeiten und ihre Organisation. Die *Systemgestaltung* vollzieht sich in Zyklen der Analyse, Synthese und Revision. Im Vordergrund stehen hierbei wechselseitige Kommunikation zwischen dem Benutzer und den Entwickler. Technisch können diese Prozesse durch *exploratives Prototyping* bei der Ermittlung von Anforderungen oder durch *experimentelles Prototyping* bei der Benutzungsgestaltung unterstützt werden. (vgl. [Pa94], S.65)

Die *Systemspezifikation* als Ergebnis der Systemgestaltung wird von den Entwicklern erstellt. Sie enthält sowohl die funktionelle Spezifikation als auch die Vorgaben für die im Einsatzbereich durchzuführenden Maßnahmen zur Qualifikation und Beschaffungen für die organisatorische und räumliche Umstellung des Einsatzbereiches. Die *Software-Realisierung* entspricht dem vom Phasenmodell bekannten *Software-Entwurf* zur Erarbeitung der inneren Architektur, der *Programmierung* und dem *Test* seiner Komponenten. Auch hier vollziehen sich Zyklen der Analyse, Synthese und Revision, die durch *Prototyping* und die Entwicklung in geeigneten *Ausbaustufen* unterstützt werden können. Das Ergebnisprodukt ist die *Systemversion*. In sie gehen das Softwaresystem und eventuell Hardwarekomponenten inklusive aller Dokumente, die von Benutzern und Entwicklern erstellt wurden ein. Die *Benutzung* der Version ergibt sich aus den regulären Arbeitsprozessen im Einsatzbereich und gleichzeitig sind die Entwickler mit der *Versionspflege*, wie Fehlerbehebung und Optimierung befaßt. (vgl. [Pa94], S.66-68)

Nachdem in diesem Kapitel die Projektgestaltung betrachtet wurde, wenden wir uns jetzt dem Softwareentwurf zu.

4 Der dialogische Softwareentwurf

Die Theorie des dialogischen Entwurfes wendet sich folgenden Fragen zu (vgl. [Pa94], S.80):

- Was ist Kooperation und unter welchen Bedingungen kommt sie zustande?
- Wie kommt eine Gruppe zu gemeinsamen Zielen? In welchem Verhältnis stehen die individuellen Ziele der Gruppenmitglieder dazu? Wann hat die Gruppe Mittel zum Erreichen der Ziele zur Verfügung?
- Wie entwickelt sich eine Gruppe?
- Was ist überhaupt Software-Entwurf?
- Welche Vorteile bietet dialogischer Software-Entwurf? Warum sind gemeinsam erstellte Entwürfe qualitativ besser als andere? Welche "Qualität" weisen diese Entwürfe auf?
- Welche Rolle spielen Modelle und Informationsmonopole dabei?
- Welche Rolle spielen Kontroversen beim Entwerfen?
- Wie entsteht ein gemeinsames Wissen in der Entwurfsgruppe, das Naur als Theorie von Programmen bezeichnet hat?
- Wie lernt man Software-Entwurf?
- Und wie kommt Innovation beim Software-Entwurf zustande?

Zu einer Entwurfssituation verabreden sich die Software-Entwickler, um über einen gewissen Zeitraum miteinander an einem Entwurf zu arbeiten. Sie bilden in diesem Zusammenhang eine Entwurfsgruppe. Diese Gruppe hat das übergeordnete Ziel, gemeinsam einen Entwurf zu erstellen. Jeder Teilnehmer bringt seine Perspektive mit und damit auch seine Ziele, Erwartungen, Interessen und erlernte Konzepte und Modelle aus der Informatik. Dieser Umstand ergibt ein mögliches Konfliktfeld zwischen den einzelnen

Teilnehmern. So gesehen müssen die Akteure beim Entwurf das Einbringen ihrer Konzepte sozial aushandeln. Das Wechselgespräch beim Entwurf ist eine Form eines argumentativen Dialoges. (vgl. [Pa94], S.82-84)

Man spricht von sozialem Handeln, wenn es ein Handeln ist, das sich in bestimmter Weise auf das Handeln anderer Akteure bezieht. Dabei kommt man zu der Fragestellung des *Fremdverstehens* anderer: Wie kann ich die Handlungsorientierung und anderer ihr geplantes Verhalten erkennen? Grundlegend ist dabei der Begriff der *Perspektive*. Die Perspektive ist der Blickwinkel eines Akteurs beim Erleben und Handeln, beeinflusst durch seinen persönlichen Bezugsrahmen. Dazu gehören beim Softwareentwurf die Ziele, Erwartungen, Interessen und erlernte Konzepte und Modelle aus der Informatik. (vgl. [Pa94], S.85)

Zu den Momenten des Handelns gehören die *Situationsdefinition* und die dafür notwendige *Kommunikation*. *Interaktion* im eigentlichen Sinne kann man in zwei ideale Grundformen unterteilen: *Kooperation* und *Austragung eines Konfliktes*. Mit Kommunikation meint man beim Entwurf die sprachlichen Mitteilungen über Inhalte, Ziele, Mittel, Absichten, Konzepte, Handlungen, die als Mittel zur Realisierung, Gestaltung und Lenkung der Interaktion selbst nötig sind. Der sozial ausgehandelte Konsens der Beteiligten über Rollen, Arbeitsmethodik, Gruppentermine, akzeptables und inakzeptables Verhalten und die Art der zu entwickelnden Interaktion kennzeichnet die Situationsdefinition beim Entwurf. Unter Interaktion werden ineinandergreifende Handlungen verstanden, die sich jeweils intentional an der Handlungsorientierung des Anderen ausrichten und eine aktive Verwirklichung von Zielen mit bestimmten Mitteln darstellen. Aufgrund der Kongruenz oder Inkongruenz der Handlungsorientierungen der Akteure, gibt es zwei idealtypische Interaktionsformen: Kooperation und Konfliktaustragung. Kooperation ist das Handeln von Akteuren mit einer gemeinsamen, übereinstimmenden Handlungsorientierung. Die zweite idealtypische Interaktionsform ist die Konfliktaustragung, d.h. die individuellen Handlungsorientierungen von wenigstens zwei Akteuren sind unvereinbar. Ein Konflikt liegt ganz klar vor, wenn die aktive Verwirklichung von Zielen des einen mit der Zielverwirklichung des anderen nicht im Einklang steht. Ändert einer der beiden seine persönlichen Ziele, so ist der Konflikt *beigelegt*. Ändern sie die gemeinsamen Ziele, so entsteht ein *Kompromiß*. Verzichten die Beteiligten auf Korrekturen ihrer Handlungsorientierung, nimmt die Konfliktaustragung die Form des *Kampfes* an. Die Akteure im dialogischen Entwurf sollten die Operation des Fremdverstehens und der Perspektivübernahmen als gemeinsame Reflexionsprozesse vollziehen. Diese lassen sich als *Argumentationsprozesse* oder *dialogische Reflexionsprozesse* beschreiben. (vgl. [Pa94], S.88-91)

Sader bestätigt die Effektivität von Kooperation. Er stellt die Frage, ob sich *Kooperation* oder *Wettbewerb* auf die Leistung der Gruppe günstiger auswirkt. Im Alltagsverständnis wird häufig angenommen, daß Wettbewerb zusätzliche Motivation schafft und daher bessere Leistung hervorrufen müßte. Empirische Untersuchungen haben aber ergeben, daß Kooperation erheblich effektiver als Wettbewerb ist. Es ist aber der Hinweis nötig, daß es keine reine Verwirklichung von Wettbewerb und Kooperation gibt: auch unter Kooperationsbedingungen gibt es Wettbewerb und unter Wettbewerbsbedingungen bleibt ein gewisses Ausmaß an Kooperation. (vgl [Sa94], S.137)

Es wurde gezeigt, daß der Dialog das zentrale Mittel ist, um zum einen den Reflexionsprozeß und zum anderen den Entwurf zu etablieren. Es gibt aber verschiedene Arten einen Dialog zu führen. Nehmen wir als Beispiel ein Arbeitsgruppe die eine Software entwerfen soll. Zur vereinbarten Konzeptbesprechung bringt ein Mitglied bereits einen ausgearbeiteten Entwurf mit, während die anderen Gruppenmitglieder sich noch keine detaillierten Gedanken gemacht haben, weil sie die Erwartung hegten, den Entwurf gemeinsam anzufertigen. Die anderen Mitglieder sehen sich jetzt in der Situation, sich mit dem vorgefertigten Entwurf auseinandersetzen zu müssen, um ihn erst einmal zu verstehen. Alle weiteren Argumentationen im Dialog, dafür oder dagegen, werden sich des Entwurfs als Verständigungsgrundlage bedienen. Dies führt dazu, daß der Entwurf immer weiter etabliert wird. Immer dann, wenn ein Dialogpartner A im Gegensatz zum Dialogpartner B über den Entwurf, also den jeweiligen Gegenstandsbereich verfügt, ist er der *modellstarke* Handelnde, er besitzt das Modellmonopol. Der *modellschwache* Partner B verfügt über keine Anteile des Gegenstandsbereiches, die nicht Teilmenge von A sind, der ja durch sein Modell den Gegenstandsbereich festgelegt hat. Der norwegische Soziologe Br? ten bezeichnet diese Situation (vgl. Abbildung 3) wird als *asymmetrischen Dialog*. Obwohl alle Akteure besten Willens kooperieren, paßt sich B immer weiter an das Modell von A an.

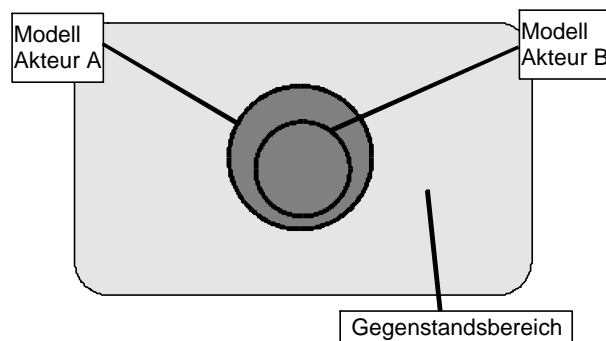


Abbildung 3: Asymmetrischer Dialog (aus [Pa94], S.94)

Broten spricht von einem symmetrischen Dialog (vgl. *Abbildung 4*), wenn die Dialogpartner *intersubjektiv ihre Perspektiven kreuzen*. Unter Perspektive versteht er eine Klasse von zusammengehörigen Sichtweisen von einem einheitlichen Blickwinkel aus auf relevante Aspekte des Gegenstandsbereichs.

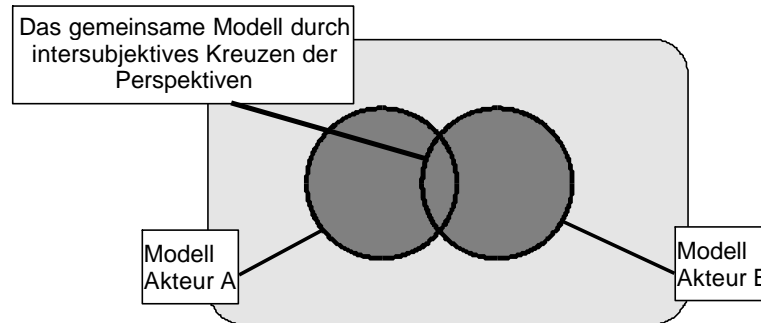


Abbildung 4: Symmetrische Dialog (aus [Pa94], S.96)

Die kleinsten Einheiten einer Argumentation sind kleinere Dialoge, wo mittels *wechselseitigen Widersprechens* und *Erwägens* bzw. von *Zustimmung* und *modifizierter Zustimmung* ein gemeinsames Ziel verfolgt und erreicht werden kann. Diese bilden den zentralen Koordinationsmechanismus für dialogische Reflexionsprozesse. Dies konstituiert die *Grundsituation des kooperativen Lernens*. Kooperatives Lernen ist eine der tragenden Säulen des dialogischen Entwurfes. Zu argumentieren lernen heißt ganz besonders, wie gerade durch wechselseitiges Widersprechen und Erwägen dennoch ein gemeinsames Ziel verfolgt werden kann. Der *argumentative Dialog* muß aber ein *symmetrischer Dialog* sein. In der Entwurfssituation wird jedes Argument potentiell von mehreren Akteuren analysiert und bewertet und es sind Gegenargumente von mehreren Seiten zu erwarten. Da die Konzepte der Beteiligten zu Beginn selten kongruent sind, werden *Kontroversen* sichtbar. Hinzu kommt auch noch eine mögliche *Rivalität* unter den Beteiligten. Konflikte, Kontroversen und Rivalität gelten in der Forschung zum Lernen in Gruppen ("kooperatives Lernen") als ein *wesentlicher Katalysator* bzw. als Vorbedingung von gemeinsamen Lernprozessen. Kontroversen können das Lernen fördern, weil sie intensive Versuche provozieren, die eigenen Gedanken und die anderer Personen klar herauszuarbeiten und möglicherweise miteinander zu vereinbaren. Mittels der Kontroversen und/oder der Rivalität befindet sich die Gruppe sachbezogen in einem *Spannungsverhältnis*. Unter diesem Druck entstehen neue Ideen, die als Argumente in den Diskurs eingebracht werden. Vorgestellte Ideen werden so ausgiebig hinsichtlich ihrer Tauglichkeit diskutiert, weiterentwickelt oder auch verworfen. In dieser Spannungssituation machen die Akteure

natürlich davon Gebrauch, schon vorhandene Konzepte auf andere Gegenstandsbereiche zu spezialisieren und als neues Argument zu bringen.

Pasch faßt dies folgendermaßen zusammen: "Um einen qualitativ hochwertigen Entwurf zu erzielen, ist idealer Weise ein ganzheitliches Verständnis jedes Gruppenmitgliedes für den Entwurf notwendig. Dieses Verständnis kommt nur in einem symmetrischen Entwurfsdialog zustande. (...) Die Kohärenz und konzeptionelle Integrität von Entwurfsentscheidungen wird durch die Vielfalt der Gegenargumente im dialogischen Entwurf entscheidend gefördert."

Eine weitere generelle Vorgehensweise, das *Brainstorming*, ließe teilweise ebenfalls beim Softwareentwurf verwenden. Beim Brainstorming werden in einer Gruppe zu einem Thema möglichst viele kreative Ideen gesammelt. Die Teilnehmer werden instruiert, möglichst viele Ideen zu einer vorgegebenen Aufgabe in der Gruppe laut und spontan zu nennen. Bewertende Aussagen sind ausdrücklich untersagt. Im allgemeinen wird nach Mitteln für die Bewältigung einer Aufgabe oder nach Konsequenzen, die sich aus einer Situation ergeben können, gesucht. Es wird angenommen, daß die Anregungen durch die Dynamik der Situation und die wechselseitige Befruchtung, Aufnahme und Weiterverarbeitung der Ideen anderer usw. zu einem Ergebnis führen.

5 Umsetzung im Softwarepraktikum

Pasch zieht seine Schlüsse als Ergebnisse aus einer Reihe von empirischen Untersuchungen, die er an der TU Berlin durchgeführt hat. Er war in den 80er Jahren wissenschaftlicher Mitarbeiter bei Christiane Floyd. Die Untersuchung fand im Rahmen eines Programmierpraktikums 1987 statt. Diese Lehrveranstaltung wurde als Programmierprojekt durchgeführt. Drei Vorlesungen dienten zur Einführung in die Problematik des dialogischen Softwareentwurfs. Das gesammelte empirische Material wurde qualitativ ausgewertet, d.h. es wurden anhand von typischen Fallbeispielen die Gruppenprozesse in Hinblick auf die Theorie beurteilt.

Da diese Fallstudie in einem ähnlichen Rahmen, wie das Softwarepraktikum an der BTU Cottbus durchgeführt wurde, wäre eine eigene Überprüfung der Theorie Paschs angebracht. Das heißt man könnte eine Gruppe fördern beim Entwurf der Software die Prinzipien des dialogischen Entwurfs zu nutzen.

Literatur

- [Pa94] J. Pasch: *Software-Entwicklung im Team*. Springer Verlag, Berlin, Heidelberg, 1994.
- [Sa94] M. Sader: *Psychologie der Gruppe*. Juventa Verlag, Weinheim, München, 1994.

Anhang A: Projektstagebuch der Gruppe: Formeleditor

In Anlehnung an die Formatierung des Tagebuchs von Daniel Dietterle werden die Beobachtungen, die für den Schwerpunkt Kooperation und Kommunikation innerhalb der Gruppe relevant sind, aber auch planungs-relevante Aspekte grau hinterlegt. Meine Interventionen, Kommentare und Vorschläge werden rot eingefärbt.

Aufgabenstellung der betreuten Gruppe: *Programmierung eines Formeleditors*

Gruppenmitglieder: Kristin Winkler, Marion Sollan, André Wallat, Wolfgang Langer

1. Treffen:

Das Treffen findet am 18.4 in der EHS von 10:00 bis 11:10. statt. Die Gruppe ist vollständig anwesend.

Zuerst erläutere ich meine Aufgabe als Betreuer. Weiterhin schlage ich vor, sich regelmäßig (einmal wöchentlich) gemeinsam zu treffen. Als Termin wird jeden Mittwoch 10 Uhr in der EHS festgelegt. Als nächstes zeigt mir die Gruppe ihre bisherige Arbeit. Es ist kein Entwurf vorhanden! Zuerst wurde nur die GUI "gehackt". Gruppe meint aber dazu, daß die GUI die zentrale Komponente ist (mir bleiben ein wenig Zweifel). Mache daraufhin den Vorschlag den Entwurf gemeinsam durchzuführen. Dies wird von der Gruppe positiv bewertet. Mein Vorschlag der Paarprogrammierung wird nicht so angenommen. Eine Planung ist zwar vorhanden, aber nicht sehr aussagekräftig! Es sind recht detaillierte Netzpläne vorhanden, aber keine Zeitplanung. Ich erläutere Zusammenhang zwischen Entwurf und Planung. Meine Vorschläge werden von der

Gruppe positiv angenommen. Es existiert keine Rollenverteilung, wie im Leitfaden vorgeschlagen! Ich dränge darauf, diese wenigstens pro forma einzuführen.

Allgemein gesagt funktioniert die Gruppe sehr gut. Kommunikation ist untereinander vorhanden. Es gibt keine dominierenden und keine unterdrückten Mitglieder in der Gruppe. Die Gruppe steht meiner Rolle als Betreuer sehr positiv gegenüber (ich kannte zwei Leute schon von früher).

2. Treffen:

Das Treffen findet am 25.4 in der EHS von 10:00 bis 11:15. statt. Die Gruppe ist nicht vollständig anwesend (ein Gruppenmitglied fehlt).

Ich lese mir die überarbeiteten und neu erstellten Dokumente durch (Entwurf, Pflichtenheft, Zeitprotokolle). Ein Entwurf ist als Klassendiagramm vorhanden (macht einen vernünftigen Eindruck). Anhand des Entwurfes wurden zwei zentrale Komponenten implementiert (Baumstruktur und GUI). Alle Gruppenmitglieder sind gleichermaßen an Design und Implementierung beteiligt (zwei Mitglieder die Baumstruktur und die anderen beiden die GUI). Eine Zeitplanung ist vorhanden, aber durch die späte Erstellung stellt sich die Frage wie geeignet diese ist. Ich dränge nochmals darauf, eine vernünftige Zeitplanung für das 2. TP am Beginn zu erstellen. erstes Teilprojekt steht kurz vor dem Abschluß, trotz sehr spät angefertigter Zeitplanung kann der Termin eingehalten werden. Ich gebe Hinweise, wie der Teilprojektsbericht aussehen sollte (anhand des Leitfadens).

Gruppe läuft weiterhin sehr gut. Rollenverteilung wird gut ausgefüllt. Keine Interventionen nötig. Die Gruppe macht einen guten Eindruck, sie wird das Projekt gut schaffen!

3. Treffen:

Das Treffen findet am 2.5 in der EHS von 10:00 bis 10:45. statt. Die Gruppe ist vollständig anwesend.

Ich sehe mir den Teilprojektsbericht an. Alle geforderten Dokumente sind erstellt. Der Entwurf ist weiterhin nur als Klassendiagramm vorhanden. Ich erläutere, daß Entwurf mehr als nur ein Klassendiagramm ist. Die Zeitplanung sieht recht gut aus, ist

wahrscheinlich aber gefaked. Zeitprotokolle sind sorgfältig geführt. Reflexionspapiere wurden von jedem Gruppenmitglied erstellt. Weiterhin diskutieren wir die Teilprojektspräsentation bei Bachmann. Er hat sich nur für das Programm interessiert und in keiner Weise für die Berichte. Ich werde mit Prof. Lewerentz die Situation besprechen. Auf Nachfrage über die Einhaltung des Codingstandards kommt raus, daß der Quelltext nicht ausreichend kommentiert ist. Ich dränge darauf sich konsequent an die Stilfibel zu halten! Eine Planung für das 2. TP ist noch nicht vorhanden. Dies wird nächste Woche besprochen.

4. Treffen:

Das Treffen findet am 9.5 in der EHS von 10:00 bis 10:35. statt. Die Gruppe ist vollständig anwesend.

Ich schaue mir Dokumentation des TP1 an. Bemängele die Dokumentation des Quelltextes und der Beschreibung der Klassendiagrammelemente (siehe letztes Treffen). Gruppe beginnt mit der Planung des 2.TP's. Ich gebe noch allgemeine Hinweise, wie diese erfolgen kann. Die Rollenverteilung wird neu diskutiert.

5. Treffen:

Das Treffen findet am 16.5 in der EHS von 10:00 bis 11:00 statt. Alle sind anwesend.

Gruppe stellt mir Planung des 2.TP's vor. Ich bemängle die zu ungenaue Zeitplanung (Ganttdiagramme). Es ist beispielsweise kein Endtermin festgelegt! Bitte die Gruppe um eine kurze Selbsteinschätzung, insbesondere wie sich die Zusammenarbeit innerhalb der Gruppe gestaltet. Es kommt aber kein richtiges Feedback, die Gruppe hat Schwierigkeiten sich selbst einzuschätzen. Bekomme später von einem Gruppenmitglied eine Einschätzung als Mail zugeschickt.

Motiviere die Gruppe, indem ich den bisherigen Projektverlauf im Vergleich zu anderen Gruppen reflektiere (nenne aber keine Namen!)

6. Treffen:

Das Treffen findet am 23.5 in der EHS von 10:00 bis 11:00. statt. Die Gruppe ist vollständig anwesend.

Ich führe Projektverfolgung durch und **bemängle das Fehlen eines Netzplanes**, welcher die Abhängigkeiten der AP's erklärt hätte. Finde weitere kleinere Fehler im Plan. **Lobe aber gleichzeitig die Aufspaltung in viele kleine AP's**. Weiterhin **erinnere ich daran, daß Reflexionspapiere, Zeitprotokolle und anderer Papierkram** bis zum 30.5. fertig sein sollen. Ich erkundige mich nach der **Kommunikation im Team, bitte um Teilnahme an der Sitzung zur Planung TP3**, um härtere Daten zur Kommunikation und Kooperation innerhalb der Gruppe zu gewinnen.

7. Treffen:

Das Treffen findet am 30.5 in der EHS von 10:00 bis 10:45. statt. Die Gruppe ist vollständig anwesend.

Ich frage nach dem Abschluß des 2.TP's. Ein lauffähiges Programm steht, der Papierkram ist noch nicht ganz fertig, sollte aber kein Problem darstellen, meint die Gruppe. Die Planung des nächsten TP's ist für Mittwoch den 6.6. angesetzt. Ich **bitte um Teilnahme zur Beobachtung und Aufnahme von "harten" Fakten**. Ich **vergleiche nochmals Zeitprotokolle, erinnere, daß die Differenz zwischen geplanter und tatsächlich gebrauchter Zeit reflektiert werden soll**.

8. Treffen:

Das Treffen findet am 6.6 in der EHS von 10:00 bis 11:20. statt. Die Gruppe ist vollständig anwesend.

Auf **Anfrage erläutere ich zunächst den Vorgang eines Reviews**. Der eigentlicher Zweck dieses Treffens ist aber die Beobachtung der Kommunikation innerhalb der Gruppe bei der Planung des 3.TP's. Ich bin passiver Beobachter, nehme keinen Einfluß und mache mir Notizen. Suche nach typischen Dialogen, die meine Vermutungen bezüglich, der

Charaktere, der Rollen innerhalb der Gruppe und der Kommunikation untereinander bestätigen. Dazu mehr in der abschließenden Gruppenreflexion.

9. Treffen:

Das Treffen findet am 13.6 in der EHS von 10:00 bis 11:00. statt. Die Gruppe ist vollständig anwesend.

Ich schaue mir die Zeitpläne (Ganttdiagramm) an und lasse sie mir erklären, um den Projektfortschritt zu beurteilen. Das Ganttdiagramm ist zwar detailliert, aber die einzelnen Pakete etwas durcheinander (bspw. AP2.3 wird von verschiedenen Personen zu verschiedenen Zeiten bearbeitet).

Ansonsten ist unser das Vorankommen sehr gut. Gruppe wird das Projekt erfolgreich abschließen.

10. Treffen:

Das Treffen findet am 4.7. in der EHS von 10:00 bis 12:15. statt. Die Gruppe ist vollständig anwesend.

Ich lasse mir die Ergebnisse des Reviews durch die Klasseneditor-Gruppe erläutern. Die Gruppe ist mit den Gutachtern nicht zufrieden. Hinweis von mir: **Wenn das Ergebnis enttäuschend ist, dann schreibt das auf! Ich erinnere erneut an das ordentliche führen von Zeitprotokollen und Besprechungsprotokollen.** Dies wurde weitestgehend getan, so daß deshalb nur noch das Gesamtreflexionspapier zu schreiben ist.

Interventionen:

Prinzipiell ist zu erwähnen, daß Kommunikation und Kooperation in dieser Gruppe gut lief. Die Gruppenmitglieder haben offen kommuniziert und ihre Probleme untereinander

besprochen. Somit war der Bedarf an Interventionen in bezüglich des Schwerpunktes Kommunikation und Kooperation nicht so groß.

Zu Beginn lagen die Probleme der Gruppe darin, nicht genau zu wissen, wie man an die Aufgabe herangehen soll. Es wurde munter losprogrammiert, ohne ein Konzept und Entwurf zu besitzen. Daraufhin schlug ich die von Pasch propagierten Methoden des gemeinsamen Entwurfs und der Paarprogrammierung vor.

Durch den gemeinsamen Entwurf soll erreicht werden, daß alle Gruppenmitglieder einen Überblick über das gesamte Projekt haben, also daß ein gemeinsames Wissen aufgebaut wird. Dieser Vorschlag wurde positiv bewertet und im weiteren Verlauf des Praktikums kontinuierlich umgesetzt. Somit hat sich meine Intervention als positiv für den Projektverlauf herausgestellt.

Da einige Gruppenmitglieder programmier-technisch nicht die stärksten waren, schlug ich weiterhin vor Programmiertätigkeiten gemeinsam (zu zweit) zu bewältigen. Dies sollte bezwecken, daß der schwächere Programmierer vom stärkeren lernt. Dieser Vorschlag wurde zuerst nicht angenommen. Jeder wollte seine Programmieraufgaben alleine lösen und nur wenn verschiedene Module integriert werden, sollte dies gemeinsam geschehen. Bei späteren Gesprächen stellte es sich aber heraus, daß zwei Gruppenmitglieder intensiver zusammenarbeiteten. Leider waren es die beiden Schwächeren, so daß der ursprüngliche Zweck der Paarprogrammierung nicht erfüllt wurde. Aufgrund der Aufgeschlossenheit der Gruppenuntereinander war es aber möglich, sich bei Problemen mit der Programmierung an andere Mitglieder zu wenden. Diese Intervention hat also nicht so gut funktioniert, was sich aber aus genannten Gründen nicht weiter negativ für den Projektverlauf ausgewirkt hat.

Was gibt es für Konsequenzen für mich, daß manche Interventionen gewirkt haben andere wiederum nicht. Generell sollte man erwähnen, daß man nur intervenieren sollte, wenn der Projekterfolg auf dem Spiel steht. Es bringt nichts, wenn man versucht jedes kleine Detail zu korrigieren oder bei kleinsten Problemen gleich massiv zu intervenieren. Man sollte der Gruppe einen gewissen Spielraum lassen, denn erstens sind gehen sie nicht planlos an das Projekt heran und zweitens gibt es nicht die EINE Lösung oder das EINZIG wahre Vorgehen in einem Entwicklungsprozeß. Das liegt vor allem an der Individualität der beteiligten Personen. Oft ist vielleicht der von der Gruppe eingeschlagene Weg, der die einzelnen Charaktere berücksichtigt, der beste Weg. Somit sind Interventionen nur sehr vorsichtig einzusetzen und nur wirklich dann wenn der Projektverlauf bzw. das Projektergebnis in Gefahr ist.

Was geschieht nun, wenn man mit einer Intervention scheitert, die man für unabdinglich für den erfolgreichen Projektverlauf erachtet. Hat man die erforderlichen Machtbefugnisse, so kann man damit die Intervention durchdrücken. Wenn nicht, muß man es auf eine argumentative Art und Weise versuchen. Das heißt insbesondere, daß man Transparenz bezüglich des eigenen Vorgehens zeigt. Gestaltet man die Gruppenbetreuung von Anfang an transparent ist, dies sicher einfacher zu bewerkstelligen. Da ich transparent war, hatte ich damit wenige Probleme.

Anhang B: Projektstagebuch der Gruppe: Khepera

In Anlehnung an die Formatierung des Tagebuchs von Daniel Dietterle werden die Beobachtungen, die für den Schwerpunkt Kooperation und Kommunikation innerhalb der Gruppe relevant sind, aber auch planungs-relevante Aspekte grau hinterlegt. Meine Interventionen, Kommentare und Vorschläge werden rot eingefärbt.

Aufgabenstellung der betreuten Gruppe: *Programmierung des Kheperas*

Gruppenmitglieder: Robert Kammerer, Andreas Keym, Christian Mönch, Maik Schubert

1. Treffen:

Das Treffen findet am 23.4 im WH3 von 11:30 bis 12:45. statt. Die Gruppe ist vollständig anwesend.

Zuerst wird ein wöchentlicher Termin festgelegt (jeden Donnerstag, 3. Block). Als nächstes sehe mir die erstellten Dokumente an. Das Pflichtenheft sieht recht gut aus. Eine Grobplanung ist vorhanden und sieht soweit gut aus. Leider ist ein Entwurf nicht vorhanden. Die Gruppe hat erst einmal angefangen loszuhacken um die Steuerung des Kheperas kennenzulernen. Diese Vorgehensweise ist einzusehen, wenn im weiteren Verlauf ein vernünftiger Entwurf erstellt wird. Zeitprotokolle sind nicht vorhanden! Ich erläutere daraufhin den Zusammenhang zwischen Entwurf und Planung und weise darauf hin, daß der Projektablauf eigentlich wichtiger als das Projekt (Programm) ist. Ich unterbreite den Vorschlag des gemeinsamen Entwurfs und der Paarprogrammierung (wird nicht sofort angenommen). Weiterhin berichte ich von eigenen Erfahrung mit dem

Khepera-Projekt. Eine Rollenverteilung ist entsprechend dem Leitfaden etabliert, wird aber wohl nur **pro forma** ausgefüllt.

2. Treffen:

Das Treffen findet am 26.4 im WH3 von 11:30 bis 12:30. statt. Die Gruppe ist vollständig anwesend.

Ich sehe mir die erstellten Dokumente an und sehe **Mängel in der Zeitplanung** (ist nicht vorhanden). Daraufhin **erläutere ich den Sinn des Softwarepraktikums als Möglichkeit Projektplanung und Projektdurchführung zu üben** (auf das Produkt kommt es nur zweitrangig an). Ich **erläutere nochmals den Zusammenhang zwischen Entwurf und Planung** und **dränge auf einen vernünftigen Projektablauf**. Die **Zeitprotokolle werden noch immer nicht vernünftig geführt!**

Sonst macht die Gruppe einen guten Eindruck, sie werden das Projekt auf jeden Fall schaffen. Kommunikation funktioniert gut. Meine Rolle als Betreuer und Berater wird akzeptiert.

3. Treffen:

Das Treffen findet am 3.5 im WH3 von 11:30 bis 12:15. statt. Die Gruppe ist vollständig anwesend.

Die Gruppe berichtet über den Projektablauf und erläutert ihre Ergebnisse. Sie macht auf das Problem mit dem Schlüssel in der EHS aufmerksam. Ich werde mit Prof. Lewerentz darüber reden. Die Gruppe spricht mich auf das Review an. Ich erläutere Sinn und Zweck dieses und verweise auf die spätere Durchführung. Wir sprechen über die Anfertigung des TP-Berichtes, ich **weise auf die Wichtigkeit der Dokumente** hin. **Ich dränge auf eine vernünftige Planung und Organisation** im 2. TP!

4. Treffen:

Das Treffen findet am 10.5 im WH3 von 11:30 bis 12:00. statt. Die Gruppe ist bis auf ein Gruppenmitglied vollständig anwesend.

Die Gruppe erläutert mir ihren TP-Bericht. Ich lese ihn mir durch und befinde ihn für gut. Das Pflichtenheft kann in einigen kleinen Punkten überarbeitet werden. Ich bemängle, daß die Planung des 2.TP als AP im 1.TP aufgeführt ist und nicht als AP1 des 2. Teilprojektes. Ich erläutere aus eigenen Erfahrungen die im Projektverlauf noch zu erwartenden Probleme.

Motiviere die Gruppe, indem ich den bisherigen Projektverlauf im Vergleich zu anderen Gruppen reflektiere (nenne aber keine Namen!). Ich muß einige kritische Äußerungen in bezug auf Sinn/Unsinn bestimmter organisatorischer/administrativer Aufgaben abwürgen.

5. Treffen:

Das Treffen findet am 10.5 im WH3 von 11:30 bis 12:00. statt. Die Gruppe ist vollständig anwesend.

Ich lasse mir von der Gruppe den bisherigen Verlauf des TP's erläutern. Sie sind laut ihrer Aussage gerade in der Implementierungs- und Testphase. Ich bin noch etwas skeptisch über den weiteren Verlauf. Ich mache die Gruppe nochmals auf die Wichtigkeit der Zeitprotokolle aufmerksam. Beim nächsten Termin soll der genaue Stand des TP's begutachtet werden.

6. Treffen:

Das Treffen findet am 23.5 in der EHS von 11:30 bis 12:15. statt. Die Gruppe ist vollständig anwesend.

Die ursprüngliche Reflexion des bisherigen Projektverlaufs konnte nicht stattfinden, da ich die Dokumente nicht erhalten hatte und auf keine bei der Besprechung vorhanden waren! Schlecht!

Ich bin der Meinung, daß die Organisation der Planung innerhalb der Gruppe doch nicht so gut läuft, wie zuvor angenommen. Planung wird zuerst nur von einer Person durchgeführt

und mit den anderen nur wenig besprochen. Bisher lief diese Variante gut, aber ich bin für die Zukunft etwas skeptisch. Die Gruppe glaubt trotz der auftretenden Probleme (eine Kalibrierungsfunktion – ursprünglich erst in TP3 geplant – muß eingebaut werden, und die Wegrückverfolgung läuft noch nicht so gut) das Teilprojekt bis nächste Woche abzuschließen; ich bin skeptisch. Ich interessiere mich für die Kommunikation innerhalb der Gruppe, will daher bei der Planung des nächsten TP's dabei sein. Die Projektgruppe äußert sich weiterhin kritisch über die Laborrechner (stürzen ab). Ich berichte über eigene Erfahrungen mit dem Khepera.

7. Treffen:

Das Treffen findet am 23.5 in der EHS von 11:30 bis 12:15. statt. Die Gruppe ist vollständig anwesend.

Das Teilprojekt konnte - wie von mir erwartet - nicht abgeschlossen werden! Es existieren noch größere Mängel bei der Labyrintherkennung (Ausgabe) und der Wegverfolgung (noch nicht fertig implementiert). Die Kalibrierungsfunktion mußte vorgezogen werden. Frage: Hätte man dies voraussehen können? Die Gruppe führt mir einige Testszenarien und die damit verbundenen Probleme vor. Ich diskutiere mit der Gruppe über Möglichkeiten, die Probleme zu beheben (auch inhaltlich). Die Gruppe bemängelt zum wiederholten male schlechte Qualität der Kheperas. Ich kann dies aufgrund der Vorführung nachvollziehen. Ich schlage vor, daß TP um 1 Woche zu verlängern, um zu versuchen die Probleme zu beheben. Sie können die technischen Probleme und den Einbau der Kalibrierungsfunktion als Gründe dafür angeben. Weiterhin erläutere ich, wie ich mir die Planung des letzten TP's vorstelle und schlage eine gemeinsame Planungssitzung vor. Die gemeinsame Sitzung stößt nicht auf ungeteilte Zustimmung!

8. Treffen:

Das Treffen findet am 7.6. in der EHS von 12:20 bis 12:45. statt. Die Gruppe ist vollständig anwesend.

Das zweite Teilprojekt wurde immer noch nicht abgeschlossen! Die Gruppe will die bis einschließlich des Wochenendes weitermachen machen und Anfang nächster Woche die nötigen Dokumente erstellen. Ich erläutere weiterhin den Ablauf des Reviews. Ich bitte

Gruppe mich zu informieren, wann die Planung des 3. TP's ansteht, um dabei zu sein und Beobachtungen bezüglich der Gruppenkommunikation zu machen ("harte" Daten).

9. Treffen:

Das Treffen findet am 14.6. in der EHS von 11:30 bis 12:15. statt. Die Gruppe ist vollständig anwesend.

Leider hat mich die Gruppe nicht über den Zeitpunkt des Treffens zur Planung des dritten TP's informiert, so daß ich daran, zwecks Datensammlung, nicht teilnehmen konnte. Ich analysiere das Planungsdokument für das 3.TP. Es ist zufriedenstellend. Ich weiß aber nicht, ob die Planung gemeinsam gemacht wurde oder, wie zuvor, nur von einer Person alleine. Ich lasse mir weiterhin das Programm im aktuellen Stand vorführen. Es lief ziemlich gut! Auch Die Gruppe berichtet mir über ihr Treffen mit dem Auftraggeber. Auch er scheint recht zufrieden mit den Ergebnissen zu sein. Das Projektteam weiß, was es jetzt noch zu tun hat, um das Projekt erfolgreich abzuschließen und bleibt in ständigem Kontakt mit dem Betreuer. Weitere termingerechte Sitzungen sind allerdings vorerst nicht mehr vorgesehen.

Interventionen:

In dieser Gruppe waren Interventionen meinerseits etwas schwieriger, da die Gruppe meinen Interventionen und Hinweisen nicht folgen braucht und dies bei einigen auch nicht getan hat. Auch hier sollte bemerkt werden, daß Kommunikation und Kooperation in dieser Gruppe gut lief.

Weiterhin lagen auch hier die Probleme der Gruppe darin, die Herangehensweise zu finden. Ohne ein Entwurf und Konzept wurde erst einmal codiert. Somit schlug ich die Methoden des gemeinsamen Entwurfs und der Paarprogrammierung vor.

Beide Vorschläge wurden abgelehnt. Der Entwurf wurde zwar irgendwie gemeinsam vollzogen, aber nicht in der Art und Weise, wie er hätte möglicherweise vonstatten gehen sollen. Paarprogrammierung wurde generell abgelehnt.

Somit beschränkten sich meine Interventionen im wesentlichen auf planerische Aspekte.

Die Konsequenzen für zukünftige Interventionen, die ich oben (bei der Formeleditorgruppe) beschrieben habe sind allgemeingültig. Deswegen sollen sie hier nicht mehr erwähnt werden.

Anhang C: Was habe ich aus dem Softwarepraktikum gelernt ?

Für mich bestand das Seminar aus zwei unterschiedlichen Teilen. Zum Einen die Literaturrecherche zu einem bestimmten Thema mit Ausarbeitung und Vortrag und zum Anderen die Arbeit mit beiden Projektgruppen.

Also beschreibe ich zunächst die Schlüsse, die ich aus meiner Ausarbeitung ziehe und die ich für mich daraus mitnehme. Wie schon erwähnt wurden bei Pasch [Pa94] der dialogische Softwareentwurf propagiert. Diese Vorgehensweise halte ich für sehr interessant, so daß man sie – wenn die Rahmenbedingungen gegeben sind – durchaus mal in einem nicht-universitären Umfeld ausprobieren könnte. Auch die Projektentwicklung in einem zyklischen Modell ist meiner Meinung nach die beste Möglichkeit ein Softwareprojekt durchzuführen. Diese Methode war mir aber schon von früheren Lehrveranstaltungen bekannt.

Da Softwareentwicklung heutzutage fast ausschließlich in Teams geschieht, ist es wichtig sich mit einer Reihe von gruppensdynamischen, psychologischen und soziologischen Phänomenen auseinanderzusetzen. Solche Phänomene werden in Saders "Psychologie der Gruppe" beschrieben.

Aus meiner Tätigkeit als Betreuer von Softwareteams habe ich gelernt, daß eine der wichtigsten "Erfolgsgaranten" für Softwareprojekte die gute Kommunikation innerhalb der Gruppe ist. Wenn die Gruppenmitglieder auf einem vernünftigen, sachlichen Level miteinander reden ist auch die unabdingliche Kooperation viel einfacher zu erreichen. In der Rolle als Gruppenleiter muß man demnach die Kommunikation fördern und als Gruppenmitglied für sich selbst und die Gruppe nutzen.