
Prüfer: Prof. Dr. rer. nat. Jochen Ludewig
Betreuerin: Dipl.-Inform. Patricia Mandl-Striegnitz

Beginn am: 09. November 1999
Beendet am: 15. Mai 2000

CR-Klassifikation: D.2.9, I.6, J.1.2, K.6.1

Diplomarbeit Nr. 1819

Quantitative Modellierung von Verhaltensaspekten in Softwareprojekten

Axel Dudler

Institut für Informatik
Universität Stuttgart
Breitwiesenstraße 20-22
D-70565 Stuttgart

Vielen Dank

an alle, die mich bei der Durchführung dieser Arbeit unterstützt haben. Vor allem natürlich an meine Betreuerin Patricia Mandl-Striegnitz, die immer und jederzeit für meine Fragen zur Verfügung stand und mir in vielen Reviewsessions unermüdlich konstruktive Kritik zukommen ließ und den rechten Weg wies. Desweiteren möchte ich mich bei meinen Eltern bedanke, die immer und immer wieder die zahlreichen Zwischendokumente und diesen Bericht korrekturgelesen haben und mich vor einr Vielzahl von Rechtschreib- und Formulierungsfehlern bewahrt haben.

Inhaltsverzeichnis

Einleitung	8
1.1 Motivation.....	8
1.2 SESAM-Grundlagen.....	9
1.2.1 Modellbeschreibungssprache	10
1.2.2 Systemarchitektur.....	11
1.3 Grundlagen des QS-Modells.....	12
1.3.1 Modellziele und -einschränkungen	12
1.3.2 Modellannahmen.....	13
1.3.3 Modellparametrisierung	14
1.3.4 Zusammenfassung	14
1.4 Aufgabenstellung	14
1.5 Vorgehensweise	14
1.6 Aufbau des Berichtes	15
2 Effekte menschlichen Verhaltens in Softwareprojekten	16
2.1 Identifikation der Aspekte	16
2.1.1 Motivation.....	17
2.1.2 Produktivität	17
2.1.3 Fluktuation.....	17
2.1.4 Beziehung zum Kunden.....	18
2.1.5 Beziehung zum Senior Management	19
2.1.6 Teamzusammenstellung	19
2.1.7 Mitarbeiterführung	21
2.2 Betrachtung der Einflüsse und Auswirkungen	23
2.2.1 Motivation.....	24
2.2.2 Produktivität	29
2.2.3 Fluktuation.....	29
2.2.4 Beziehung zum Kunden.....	32
2.2.5 Beziehung zum Senior Management	33
2.2.6 Teamzusammenstellung	34
2.2.7 Mitarbeiterführung	35
3 Umsetzung der Aspekte	37
3.1 Erarbeitung geeigneter Kriterien zur Auswahl der Aspekte.....	37
3.2 Motivation der Entwickler.....	39
3.2.1 Gesamtmotivation der Entwickler.....	41
3.2.2 Mitarbeitergespräche.....	43
3.2.3 Motivationsmaßnahmen.....	43
3.3 Produktivität.....	44
3.4 Fluktuation.....	44
3.4.1 Kündigung.....	45
3.4.2 Unvorhersehbare Kündigung.....	45
3.5 Beziehung zum Kunden.....	46
3.5.1 Projektstatus (Kunde).....	46

3.5.2	Motivation des Kunden	46
3.5.3	Statusmeetings mit dem Kunden	47
3.6	Beziehung zum Senior Management	47
3.7	Teamzusammenstellung.....	48
3.7.1	Veränderliche Erfahrungen	48
3.7.2	Fehlzeiten.....	49
3.8	Mitarbeiterführung.....	50
3.8.1	Überstunden.....	50
3.9	Anforderungsänderungen.....	50
4	Quantifizierung des Modells	51
4.1	Motivation.....	52
4.2	Art der Tätigkeit.....	54
4.3	Mitarbeitergespräch	55
4.4	Veränderliche Erfahrungen	55
4.5	Überstunden	56
4.6	Unterbeschäftigung	57
4.7	Motivationsmaßnahmen.....	57
4.8	Produktivität.....	58
4.9	Fehlzeiten.....	59
4.10	Kündigung durch schlechte Motivation.....	61
4.11	Unvorhersehbare Kündigung	61
4.12	Projektstatus.....	62
4.13	Motivation des Senior Managements	65
4.14	Motivation des Kunden.....	66
5	Test und Nachweis ausgewählter Aspekte	67
5.1	Vergessen von Motivationseffekten	68
5.2	Motivationsmaßnahmen heben die Motivation	68
5.3	Die Art der Tätigkeit beeinflusst die Motivation	70
5.4	Veränderliche Erfahrungen.....	71
5.5	Überstunden	72
5.6	Die Motivation beeinflusst die Produktivität.....	74
5.7	Fehlzeiten sind von der Motivation abhängig.....	75
5.8	Kündigung durch zufällige Ereignisse.....	76
5.9	Schlechte Motivation führt zu Kündigung.....	77
6	Bewertung und Ausblick	78
6.1	Ein persönlicher Rückblick.....	78
6.2	Was kommt danach?	80
7	Literaturverzeichnis	81

Abbildungsverzeichnis

<i>Abbildung: 1: Architektur des SESAM-Systems</i>	<i>11</i>
<i>Abbildung: 2: Modellierung der Zusammenhänge nach Herzberg(1958)</i>	<i>39</i>
<i>Abbildung: 3: Verkürzte Modellierung der Zusammenhänge</i>	<i>40</i>
<i>Abbildung: 4: Anpassungsfunktion für die Motivation bei positiven Änderungen</i>	<i>52</i>
<i>Abbildung: 5: Anpassungsfunktion für die Motivation bei negativen Änderungen</i>	<i>53</i>
<i>Abbildung: 6: Anpassungsfunktion der Produktivität in Abhängigkeit zur Motivation</i>	<i>58</i>
<i>Abbildung: 7: Testprotokoll Vergessen von Motivationsereignissen</i>	<i>67</i>
<i>Abbildung: 8: Testprotokoll Teamschulung</i>	<i>68</i>
<i>Abbildung: 9: Testprotokoll Mittagessen.....</i>	<i>69</i>
<i>Abbildung: 10: Testprotokoll Auswirkungen der Art der Tätigkeit auf die Motivation</i>	<i>70</i>
<i>Abbildung: 11: Testprotokoll veränderliche Erfahrungen.....</i>	<i>71</i>
<i>Abbildung: 12: Testprotokoll Überstunden.....</i>	<i>72</i>
<i>Abbildung: 13: Testprotokoll Überstunden.....</i>	<i>73</i>
<i>Abbildung: 14: Testprotokoll Produktivität</i>	<i>74</i>

Tabellenverzeichnis

<i>Tabelle: 1: Arbeitsmotivationsfaktoren nach Herzberg [Herzb59] (nach Wichtigkeit sortiert)</i>	<i>23</i>
<i>Tabelle: 2: Auswirkungen besonderer Zufriedenheit bzw. besonderer Unzufriedenheit nach Herzberg (nach Auftretenshäufigkeit geordnet)</i>	<i>27</i>
<i>Tabelle: 3: Auswirkung der aktuellen Tätigkeiten auf die Erfahrungen der Mitarbeiter</i>	<i>48</i>
<i>Tabelle: 4: Einteilung der Motivation in Klassen</i>	<i>51</i>
<i>Tabelle: 5: Motivationsänderung nach Erfahrungsklassen</i>	<i>54</i>
<i>Tabelle: 6: Motivationsänderung je Motivationsmaßnahme</i>	<i>56</i>
<i>Tabelle: 7: Beispiele für Produktivitätsänderungen je nach Motivation</i>	<i>58</i>
<i>Tabelle: 8: Wahrscheinlichkeiten für die Länge der Fehlzeiten</i>	<i>59</i>
<i>Tabelle: 10: Theoretischer Aufwand und Zeitbedarf für einzelne Projektphasen</i>	<i>61</i>
<i>Tabelle: 11: Auswirkung der Motivation des Senior Managements auf den Abstand zwischen zwei Statusmeetings</i>	<i>65</i>
<i>Tabelle: 12: Testprotokoll Fehlzeiten</i>	<i>75</i>

Kapitel 1:

Einleitung

1.1 Motivation

Softwareprojekte erfolgreich durchzuführen ist eine schwierige Aufgabe. Als Projektleiter wird man immer aufs Neue mit einer Vielzahl von Problemen und kritischen Situationen konfrontiert, die es zu meistern gilt. Der Projektleiter als zentrales Steuerungsorgan ist daher ein wichtiger Faktor in allen Softwareprojekten. Um so erstaunlicher ist es, daß der Ausbildung von Projektleitern wenig Beachtung geschenkt wird und so deren Qualifikationen und Erfahrungen in der Praxis erschreckend gering sind [Mandl97].

In der Abteilung Software Engineering der Universität Stuttgart wird daher ein neuer Ansatz entwickelt, mit dessen Hilfe angehende und bereits eingesetzte Projektleiter ausgebildet werden können. Im SESAM-Projekt (**S**oftware **E**ngineering **S**imulation durch **A**nimierte **M**odelle) wird ein System zur interaktiven Simulation von Softwareprojekten geschaffen. In diesem System werden alle Objekte eines Softwareprojektes und deren Zusammenhänge beschrieben, so daß ein Spieler die Rolle des Projektleiters übernehmen und ein Softwareprojekt am Bildschirm durchführen kann. Durch den dynamischen Aufbau des Systems reagieren alle Objekte nach den Anweisungen des Spielers, so daß es von seinem Geschick abhängt, ob ein Projekt mehr oder weniger erfolgreich durchgeführt wird. Die Hypothese hinter diesem Ansatz ist, daß die Spieler so praxisnahe Erfahrungen sammeln können, ohne dafür reale Projekte verwenden zu müssen. Außerdem wird die Motivation und den Lernerfolg durch eine spielorientierte Lernmethode hoch gehalten. Im SESAM-System wurde ein erstes Simulationsmodell, das sogenannte QS-Modell geschaffen [Drapp00]. Das Modell bietet dem Spieler eine Vielzahl von Vorgehensweisen an und

zeigt ihm direkt, welche Auswirkungen seine Handlungen auf den weiteren Spielverlauf haben und welchen Schaden er durch falsche Entscheidungen anrichten kann.

Das Hauptziel von SESAM ist es, die Ausbildung von Projektleitern durch eine Simulationskomponente zu ergänzen. Hierfür wurde ein Schulungskonzept entwickelt [Mandl00], das diese Simulation mit Schulungen und Feedbackgesprächen zu einer kompletten Projektmanagement-Ausbildung formt. Dieses Konzept wurde bereits erprobt und hat bestätigt, daß die Integration einer Simulationskomponente durchweg positive Ergebnisse liefert.

Die Ergebnisse des ersten Experimentes wurden ausgewertet und haben Ideen für Erweiterungen des SESAM-Systems hervorgebracht. So wird beispielsweise an ein Modell für Fortgeschrittene gedacht, das neben den planbaren und vorhersehbaren Aspekten eines Softwareprojektes auch Effekte beinhaltet, die nicht planbar oder sogar zufällig sind oder beispielsweise auf autonomem Verhalten von Mitarbeitern beruhen. Im aktuellen QS-Modell sind diese Effekte noch nicht integriert.

Bedenkt man die Tatsache, daß es in realen Softwareprojekten oft genau diese Effekte sind, die Projektleiter fordern und ihm Probleme bereiten, so stellt sich die Forderung, diese Aspekte in das QS-Modell zu integrieren. Dies bedeutet einen zusätzlichen Realitätsgewinn für die Simulation und zeigt dem Spieler bereits während seiner Ausbildung, mit welchen komplexen und teilweise unüberschaubaren Problemen er in der Realität zu rechnen hat.

1.2 SESAM-Grundlagen

Bereits im Jahre 1989 reifte in der Abteilung Software Engineering an der Universität Stuttgart die Idee, Studierenden und angehenden Projektleitern die Anwendung der Methoden und Werkzeuge des Software-Engineerings in der Praxis mit Hilfe eines wissenschaftlich fundierten Computerspieles beizubringen. Elf Jahre später ist aus dieser Idee ein mittlerweile fertiges System entstanden, das seine Tauglichkeit bereits in mehreren Experimenten bewiesen hat.

Die wichtigste Aufgabe im Rahmen des SESAM-Projektes war es, eine Simulationsumgebung zu schaffen, in der Spieler üben können, ein Softwareprojekt durchzuführen. Diese Simulationsumgebung muß flexibel genug sein, um verschiedene Prozeßmodelle und Schwierigkeitsgrade zu unterstützen.

Softwareprojekte werden normalerweise durch Prozeßdefinitionen modelliert. SESAM ist im Gegensatz dazu ein regelbasiertes System. Mit Hilfe von Regeln, Relationen und Entitäten wird ein gesamtes Projektmodell realisiert. Der eigentliche Simulator (hier die "Basismaschine" genannt), überprüft nun in regelmäßigen Abständen welche Regeln aufgrund ihrer Vorbedingungen feuern und führt sie dann aus. Es werden also mögliche Effekte und Zusammenhänge eines Softwareprojektes modelliert, die einen Rahmen für ein Projekt darstellen. Der Projektlauf selbst ergibt sich aus dem Zusammenspiel dieser

Komponenten. All diese Regeln arbeiten auf einer globalen Datenstruktur, die als Graph repräsentiert wird.

1.2.1 Modellbeschreibungssprache

Um modellieren zu können wird natürlich eine regelbasierte Modellierungssprache benötigt. Eine genaue Sprachdefinition wird hier nicht gegeben, aber ein kleiner Eindruck soll dennoch vermittelt werden. Zum besseren Verständnis ist hier ein kurzes Demonstrationsmodell angegeben [aus Drapp98]:

```

model Example is
  entity type Developer is
    attributes
      costs: real; || $/day
      is_assigned: boolean;
      experiences: Experience_Record_Type;
  end type;

  entity type Specification supertype
    Document is ...
  end type;

  relation type worksOn is
    roles
      who: Developer;
      what: Document;
    attributes
      productivity: real; || AFP/day
  end type;

  command let_specify (a_Dev: Developer)
    structure
      the_Spec: Specification;
    constraints
      not exists (worksOn (a_Dev, the_Spec));
    action_part
      create relation worksOn with
        who := a_Dev;
        what := the_Spec;
        productivity := give_prod_for (a_Dev);
      end create;
      send_message Begins_to_specify(a_Dev);
  end command;

  discrete_rule Developer_specifies
    structure
      a_Dev: Developer;
      the_Spec: Specification;
      specifies: worksOn (a_Dev, the_Spec);
    constraints
      ...
    action_part
      ...
  end rule;

```

```
begin at 1998/06/22/8:00 timestep ...
  create entity D1: Developer aka "J. Smith" with
    costs := 1000.00;
    is_assigned := TRUE;
    experiences := ...
  end create;

  create entity S: Specification
    aka "The Specification" with ...
  end create;

end model;
```

Modelle bestehen hauptsächlich aus drei großen Teilen:

Schemamodell

Im Schemamodell werden alle Entitäts- und Relationstypen definiert, die im weiteren Verlauf benötigt werden (z.B.: die Entität "Developer", also der Entwickler, mit den Attributen, Kosten und Erfahrungen und einer Kennzeichnung, ob er für das Projekt eingestellt ist)

Regelmodell

Im Regelmodell werden alle Regeln definiert, die die Basismaschine während eines Spieles prüfen soll (z.Bsp. die Regel "Developer_Specifies", die in jedem Spielzug für jeden Entwickler feuert, der die Spezifikation bearbeitet).

Situationsmodell

Das Situationsmodell beschreibt die Ausgangssituation des Projektes. Hier werden Variablen vordefiniert und es wird ein bestimmter Projektzustand vorgegeben (wie z.Bsp. ein zur Verfügung stehender Entwickler mit dem Namen J.Smith).

1.2.2 Systemarchitektur

SESAM ist als dreistufige Architektur aufgebaut (Abbildung 1). Auf der obersten Ebene befindet sich der Modellbauer, der das Modell mit Hilfe eines Texteditors erstellt und dann durch einen Compiler in eine Form übersetzen läßt, die die Basismaschine ausführen kann.

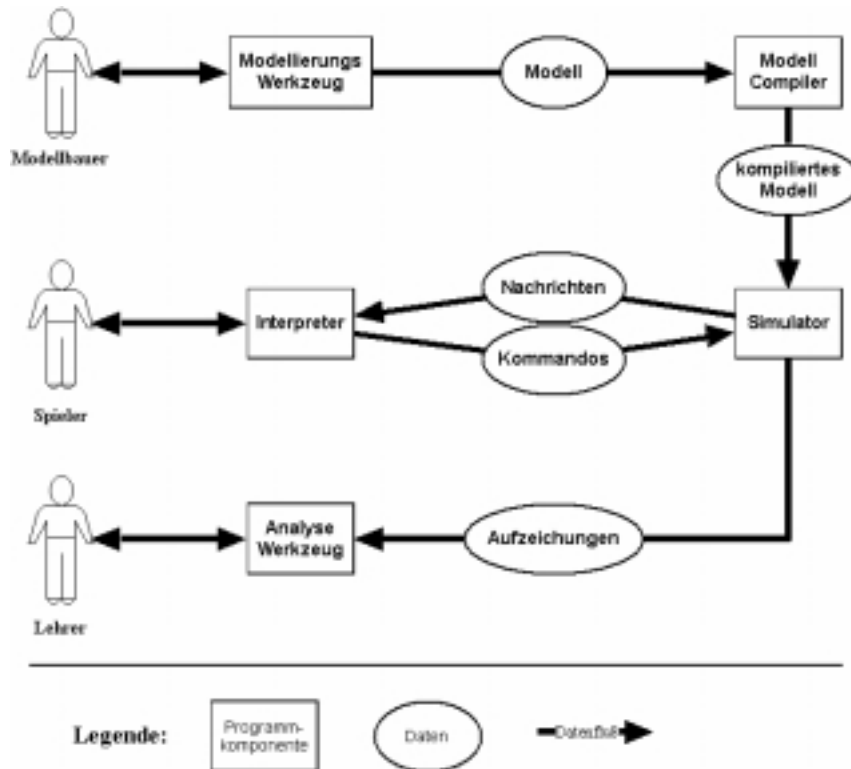


Abbildung 1: Architektur des SESAM-Systems

Auf der zweiten Ebene befindet sich der Spieler, der mit Hilfe des Interpreters Kommandos zur Steuerung des Projektes abgeben kann. Diese werden an den Simulator weitergegeben. Der Simulator vergleicht dann mit jedem Zeitschritt alle Regeln mit der aktuellen Situation und feuert Regeln, deren Bedingungen erfüllt sind. Auf der dritten Ebene befindet sich der Tutor, der mit Hilfe von Auswertungstools die Aufzeichnungen, die der Simulator während der Ausführungsphase gespeichert hat, detailliert auswerten kann.

1.3 Grundlagen des QS-Modells

Nachdem die Grundlage geschaffen war, mußte nun ein geeignetes Modell entworfen werden, um das System zu komplettieren. Dieses Modell wird in diesem Kapitel kurz vorgestellt.

1.3.1 Modellziele und -einschränkungen

Vor der Entwicklung des Modells wurde eine ausführliche Anforderungsanalyse durchgeführt, um die Ziele des Modells zu definieren. Drappa [Drapp00] legte für dieses Modell folgende Basisziele fest:

- *Flexibilität:* Das Modell muß flexibel sein, um die Spieler nicht in ihrer Entscheidungsfreiheit einzugrenzen. Der Spieler soll eigenständig eine Lösung für ein Problem entwickeln, anstatt zielgerichtet auf eine Lösung hingeführt zu werden.
- *Detailtreue:* Befehle, die der Spieler eingibt, sollen den Aktivitäten, die ein Projektleiter in der Realität durchzuführen hat stark ähneln, um die erlernten Erfahrungen einfach in die Realität übernehmen zu können.

- *Umfassend*: Das Modell soll alle Phasen eines Softwareprojektes umfassen. Vor allem in späteren Phasen eines Projektes werden die Fehler gefunden, die in frühen Phasen gemacht wurden. Der Spieler soll mit diesem ernstesten Problem konfrontiert werden.
- *Empirische Fundierung*: Das Modell soll auch dazu dienen, dem Spieler Erkenntnisse aus empirischen Studien nahezubringen. So kann erste "Praxiserfahrung" vermittelt werden.

Nach ersten Untersuchungen wurde jedoch schnell klar, daß es nicht möglich sein wird, ein universelles Modell für alle Arten von Projekten zu schaffen. Aus diesem Grund wurde das Modell auf bestimmte Projektklassen beschränkt [Drapp00]:

- *Projektklasse*: Vertragssoftware
- *Organisationsform*: Projektorganisation
- *Vertragsform*: Festpreisvertrag
- *Projektgröße*: 100-1000 Function Points
- *Größe des Projektteams*: etwa 5 Entwickler
- *Anwendungsgebiet und Art der Applikation*: Hier wird der Gültigkeitsbereich nicht allzu sehr eingeschränkt. Das Modell soll aber vor allem für Informations- und Verwaltungssysteme gültig sein.
- *Entwicklungssituation*: Die Anforderungen des Kunden sind stabil. Die Entwickler sind mit dem Anwendungsgebiet und der Entwicklungsumgebung vertraut.

Eine Einschränkung bezüglich des Prozeßmodells soll nicht bestehen. Es soll dem Spieler überlassen werden, welche Art der Vorgehensweise er wählt. Da es aber nicht möglich war, alle Vorgehensweisen zu unterstützen, wurde festgelegt, daß auf jeden Fall eine Entwicklung nach dem Wasserfallmodell möglich sein sollte. Diese sollte auch zu den besten Ergebnissen führen.

1.3.2 Modellannahmen

Das QS-Modell hat den Zweck, dem Spieler möglichst viele Effekte aus realen Projekten nahezubringen. Hierfür mußten Effekte ausgewählt werden, die in das Modell integriert wurden. Nach einem Auswahlprozeß wurden folgende Haupteffekte definiert [Drapp98]:

- Die Produktivität eines Entwicklers sinkt, wenn die Anzahl der Entwickler im Projekt ansteigt.
- Die Erfahrungen und Kenntnisse eines Entwickler beeinflussen seine Produktivität essentiell.
- Die Vollständigkeit und Qualität eines Produktes hängt von den Ergebnissen früherer Projektphasen ab.
- Reviews ermöglichen es, Fehler bereits in frühen Projektphasen zu finden. Die Vollständigkeit und Korrektheit der Vorgängerdokumente beeinflussen die Effektivität dieser Reviews.
- Durch Tests können Fehler im Code gefunden werden.
- Je später im Entwicklungsprozeß ein Fehler gefunden wird, um so höher sind die Kosten, ihn zu entfernen.

1.3.3 Modellparametrisierung

Die Parametrisierung des Modells ist ein kritischer Punkt. Ein Ziel des Modells war es, eine möglichst breite, empirisch fundierte Datenbasis zu erhalten, anhand derer die Effekte im QS-Modell quantifiziert werden können. Empirische Studien sind aber auf dem Gebiet des Software Engineerings sehr dünn gesät. So wurden die Hauptteile des Modells mit Hilfe der Studien von Jones [Jones96] parametrisiert. Zur Validierung der Daten wurde vor allem die COCOMO-Studie von Boehm [Boehm86] herangezogen.

1.3.4 Zusammenfassung

Das Modell wurde in mehrjähriger Arbeit komplett erstellt, quantifiziert und auch validiert. Seine Tauglichkeit für die Ausbildung hat es bereits in mehreren Experimenten bewiesen [Mandl00]. Aus den Erkenntnissen dieser Experimente entstanden Ideen für weitere Verbesserungen an dem System, die in dieser Arbeit aufgegriffen werden.

1.4 Aufgabenstellung

Die Aufgabenstellung wurde in [Mandl99a] definiert:

“Ziel dieser Diplomarbeit ist es, das QS-Modell um (Zufalls)Aspekte im Verhalten von Mitarbeitern und Kunden zu erweitern. Beispielsweise sollen Aspekte wie die Motivation der Mitarbeiter, die Fluktuation von Mitarbeitern oder Anforderungsänderungen durch den Kunden zur Laufzeit des Projekts modelliert werden. Zuerst müssen diese Aspekte und vor allem ihre Effekte im Projekt identifiziert werden. Die Zusammenhänge im Projekt müssen beschrieben (wie wirkt sich beispielsweise die Motivation eines Entwicklers auf seine Produktivität aus und wodurch wird wiederum die Motivation eines Entwicklers beeinflusst) und quantifiziert werden. Nachdem die Aspekte und ihre Auswirkungen auf den Projektverlauf quantifiziert sind, müssen sie im QS-Modell umgesetzt werden. Es müssen geeignete Metriken und Gleichungen bestimmt werden, um diese zusätzlichen Effekte zu modellieren.

Weil der Kenntnisstand über die quantitativen Zusammenhänge noch sehr unzureichend ist, muß sich folglich auf Hypothesen gestützt werden, falls geeignete empirische Daten fehlen. Aus diesem Grund muß anschließend durch geeignete Testszenarien nachgewiesen werden, daß die implementierten Effekte insgesamt plausibel sind und die Spielverläufe plausible Resultate liefern.”

1.5 Vorgehensweise

Um diese Aufgabe zu bewältigen, wurde ein systematischer Lösungsansatz gewählt. Zuerst mußte ein Überblick über die Literatur im Themenbereich geschaffen werden. Es wurden empirisch oder theoretisch belegte Aspekte gefunden, die den Verlauf eines Softwareprojektes beeinflussen. Hieraus wurden diejenigen Aspekte ausgemacht, die aufgrund von autonomen, zum Teil nicht planbaren Verhalten von Mitarbeitern und Kunden entstehen. Aus der Vielzahl der in der Literaturrecherche gefundenen Aspekte wurde dann eine Auswahl getroffen. Es wurden diejenigen Aspekte festgelegt, die die größte Auftrittswahrscheinlichkeit besitzen oder durch die Literatur als extrem wichtig definiert werden. Bei dieser Auswahl wurde bereits auf das Ziel der Arbeit vorausgeschaut. Es sollten nur solche Aspekte für die Weiterverarbeitung Verwendung finden, die auch eine klar definier-

bare Auswirkung auf ein Softwareprojekt haben und voraussichtlich im Rahmen des QS-Modells realisierbar und quantifizierbar sind.

In einer zweiten Literaturrecherche wurden im Anschluß daran allgemeine Zusammenhänge und Einflüsse der in der ersten Literaturrecherche definierten Verhaltensaspekte in einem Softwareprojekt erforscht. Die Fragen, die sich hier stellten lauteten: Wie oder wodurch wird ein solcher Aspekt beeinflusst und welche Auswirkungen hat er auf weitere Aspekte des Gesamtprojektes? Welche Abhängigkeiten zwischen einzelnen Aspekten lassen sich eventuell modellieren? Die allgemein definierten Zusammenhänge wurden nun an das QS-Modell von SESAM angepaßt und an Stellen, an denen nicht genügend Zusammenhänge verfügbar waren durch Hypothesen ergänzt. Hierbei wurde ein Metriksystem für die genauen Zusammenhänge der einzelnen Aspekte innerhalb des QS-Modells definiert. Es enthält, noch unparametrisiert, alle Einflüsse und Auswirkungen auf konkrete Regeln und Variablen im QS-Modell. Dies war außerdem der letzte Auswahlschritt für die Aspekte und ihre Zusammenhänge. Hierfür wurden Kriterien definiert, anhand derer eine Auswahl der Aspekte für die Umsetzung im QS-Modell erfolgt ist. Aspekte, die sich nicht im QS-Modell definieren ließen, wurden in diesem Schritt eliminiert. Alle Aspekte und Zusammenhänge, die nach diesem Schritt noch vorhanden waren, wurden im Modell implementiert.

In einem nächsten Schritt wurden die modellierten Zusammenhänge nun parametrisiert. Hierzu wurden geeignete Berechnungsvorschriften für die definierten Metriken gefunden und parametrisiert. Dies erfolgte durch eine Literaturrecherche. Da für einzelne Aspekte keine oder nur sehr wenige empirischen Daten vorhanden waren, mußte die Quantifizierung teilweise aufgrund von Hypothesen vorgenommen werden.

Die quantifizierten Aspekte und Zusammenhänge wurden daraufhin in das QS-Modell von SESAM-2 umgesetzt und durch einen Modultest verifiziert.

Ein weiterer Punkt bei der Erstellung eines quantitativen Modells ist eine Verifikation. Durch Testfälle wurde verifiziert, daß die implementierten Aspekte im Verlauf einer Simulation die gewünschten Effekte erzielen. Für einige wenige Effekte wurde auch eine Validierung durchgeführt. Besonderer Wert wurde hierbei auf aufgestellte Hypothesen gelegt. Da es für die meisten realisierten Aspekte nur eine sehr dünne Datenbasis gibt, konnte in vielen Fällen keine vollständige Validierung durchgeführt werden, da Zweit- und Drittquellen nicht vorhanden waren. In diesen Fällen wurden dann einfache Plausibilitätsprüfungen vorgenommen.

1.6 Aufbau des Berichtes

Dieser Bericht orientiert sich in seinem Aufbau stark an der Vorgehensweise während der Diplomarbeitsphase. In Kapitel 2 werden Effekte menschlichen Verhaltens anhand aktueller Literatur beschrieben und die dazugehörigen Zusammenhänge identifiziert. Kapitel 3 beinhaltet dann die Auswahl und Modellierung einzelner Aspekte, die in das QS-Modell integriert werden sollen, anhand von zuvor definierten Kriterien. In Kapitel 4 wird die

Parametrisierung der ausgewählten Aspekte beschrieben. Kapitel 5 zeigt einen Ausschnitt der Validierung bzw. der Plausibilitätsprüfung, die abschließend erfolgte. In Kapitel 6 wird die Arbeit abschließend kritisch betrachtet und es wird ein Ausblick auf weitere Tätigkeiten gegeben. Abgeschlossen wird dieser Bericht durch ein Literaturverzeichnis.

Kapitel 2:

Effekte menschlichen Verhaltens in Softwareprojekten

In der Vergangenheit wurden viele Studien durchgeführt, die Risiken und Schwierigkeiten innerhalb von Projekten aufzudecken versuchten und dies auch sehr erfolgreich geschafft haben. Es werden jedoch in diesen Studien oft nur die technischen Aspekte der Softwareentwicklung betrachtet, wie beispielsweise die Kosten- und Zeitplanung oder die Auswahl der geeigneten Werkzeuge. Ein Punkt, der oft vernachlässigt wird, sind “Soft-Factors”, menschliche Aspekte. Diese Einflußfaktoren auf Projekte entstehen allein dadurch, daß Menschen autonom und teilweise unplanbar agieren. In einigen wenigen Studien wurde bereits bewiesen, daß diese Aspekte keinesfalls vernachlässigbar sind, sondern vielmehr einen beträchtlichen Einfluß auf den Verlauf von Softwareprojekten haben können.

In diesem Kapitel wird aufgrund vorhandener Literatur ein Überblick über diese “Soft-Factors” gegeben. Hierfür wurden diese Aspekte zuerst identifiziert. Im nächsten Schritt wurden alle Zusammenhänge der identifizierten Aspekte in Softwareprojekten betrachtet. Dieser Schritt war nötig, um im späteren Verlauf der Arbeit eine Auswahl treffen zu können, welche Aspekte in das QS-Modell übernommen werden können und welche nicht.

2.1 Identifikation der Aspekte

Dieses Kapitel beschreibt Aspekte, die ein Softwareprojekt beeinflussen und aufgrund von autonomem, zum Teil nicht planbarem Verhalten von Mitarbeitern entstehen. Die hier angeführten Aspekte sind diejenigen, die in der aktuellen Literatur als die wichtigsten

angesehen werden. Alle Aspekte werden in einer Systematik erfaßt, um diese später in das QS-Modell einzubinden. Die einzelnen Aspekte werden erläutert und die Zusammenhänge dieser Aspekte innerhalb von Projekten kurz angesprochen. Eine ausführlichere Definition der Zusammenhänge und deren Umsetzung in das QS-Modell erfolgen zu einem späteren Zeitpunkt (siehe Kapitel 3).

2.1.1 Motivation

Die Motivation kann ohne Zögern als der zentrale Verhaltensaspekt in jeder Art von Entwicklungsprojekten angesehen werden [Elzer89] [Weltz92] [Thaye97] [Powel97]. Eine noch größere Rolle kommt ihr in Softwareprojekten zu, da sich hier erwiesenermaßen eine enorme Abhängigkeit zur Produktivität darstellt [Couga78]. Hochmotivierte Mitarbeiter in einem Projekt zu haben, ist das große Ziel aller Führungskräfte. Jedoch kann die Motivation nicht als isolierter Aspekt angesehen werden, da sie mit allen anderen identifizierten Aspekten in engem Zusammenhang steht. So wird bspw. durch eine hohe Fluktuationsrate die Motivation gesenkt, was wiederum eine noch höhere Fluktuationsrate nach sich zieht. Hier entsteht also ein regelrechter Teufelskreis. Kurz zusammengefaßt läßt sich sagen, daß eine hohe Motivation aus der Befriedigung aller Bedürfnisse des Mitarbeiters resultiert [Herzb59] [Maslo54]. Da sich der Faktor Motivation seine Komplexität hauptsächlich aus dem Zusammenspiel mit anderen Aspekten bezieht, wird eine genauere Definition in Kapitel 2.2 erstellt.

2.1.2 Produktivität

Eine hohe Produktivität ist eines der großen Ziele, das Führungskräfte erreichen möchten. Projekte, die mit hochproduktiven Mitarbeitern durchgeführt werden, können schneller, billiger und erfolgreicher beendet werden. Doch ähnlich dem Faktor "Motivation" ist auch der Faktor "Produktivität" von vielen anderen Faktoren abhängig. So spielt hier beispielsweise die Motivation, die Arbeitsumgebung und die Erfahrung eine große Rolle.

2.1.3 Fluktuation

Die Fluktuation bezeichnet den Zu- und Abgang der Mitarbeiter in einem Projekt. Die Mitarbeiterzahl kann sowohl durch Zugänge erhöht, als auch durch Abgänge erniedrigt werden. Ein weiterer wichtiger Punkt innerhalb dieses Aspektes ist die Kontinuität der Besetzung einer Projektgruppe.

Die Auswirkungen von Fluktuation in einem Projekt sind nicht einfach zu definieren. So ist es einerseits von Vorteil, wenn beispielsweise "frisches Blut" in das Projekt kommt um alte und eingefahrene Standards zu überdenken, aber andererseits geht durch einen ausscheidenden Mitarbeiter auch viel an Wissen über das Projekt verloren.

Ausscheiden aus dem Projekt

Das Ausscheiden eines Mitarbeiters aus einem Projekt muß als einer der wichtigsten Problemfälle angesehen werden [Metzg96] [Couga78] [Barto97].

Das Ausscheiden von Mitarbeitern kann in zwei Fälle unterteilt werden. Einerseits kann es nur temporär sein. Dies trifft beispielsweise bei Urlaub oder Krankheit zu. In diesem Fall muß die Arbeitskraft der betreffenden Person nur temporär ersetzt werden oder die zu erledigenden Aufgaben müssen verschoben werden. Diese beiden Fälle können durchaus als nicht planbar angesehen werden, da sich eine Krankheit nur selten voraussagen läßt

und auch Urlaub oft kurzfristig eingereicht wird. Das Argument, ein Urlaub sollte immer im voraus geplant werden und ist damit lange vorhersehbar, kann hier nicht akzeptiert werden. Ein nicht genehmigter Urlaub, und sei er noch so kurzfristig beantragt, führt oft zu einem nicht zu vernachlässigbaren Motivationsschwund.

Ein dauerhaftes Ausscheiden von Mitarbeitern bringt andererseits oft größere Probleme mit sich. Die Erfahrung und das Wissen über das Projekt sind nur sehr schwer zu ersetzen. Eine Einarbeitung eines Mitarbeiters, sei es nun extern oder intern, ist ein kostspieliger und zeitraubender Vorgang. Er bindet auch Ressourcen der anderen Projektmitarbeiter, die den Einsteiger bei der Einarbeitung unterstützen müssen.

Es besteht des weiteren die Gefahr, daß ein Mitarbeiter der eine Schlüsselposition (s.u.) bekleidet, das Projekt verläßt. In einem solchen Fall vervielfachen sich die angesprochenen Probleme sogar meistens noch. In der heutigen Zeit kommt noch erschwerend hinzu, daß es oft nicht möglich ist, freigewordene Stellen äquivalent zu besetzen, da der Arbeitsmarkt im Bereich EDV sehr umkämpft und leergefegt ist.

Das Ausscheiden eines Mitarbeiters kann aber auch heilsam sein. So wirkt sich das Ausscheiden eines unproduktiven "Mitläufers" durchaus positiv auf die Motivation des Teams aus [Davis97].

Hinzufügen neuer Mitarbeiter

Auch das Hinzufügen von Mitarbeitern ist nicht problemlos. Es existieren nicht nur die Schwierigkeiten, überhaupt den richtigen Projektmitarbeiter zu finden [Rothm99], sondern es muß auch auf die Problematik Rücksicht genommen werden, daß einem Projekt nicht unbegrenzt Mitarbeiter hinzugefügt werden dürfen. Es wurde bewiesen, daß ab einer bestimmten Anzahl von Mitarbeitern der Kommunikations- und Einarbeitungsaufwand überhand nimmt ("Adding manpower to a late project makes it even later") [Brook74].

Kontinuität der Besetzung

Die Kontinuität der Besetzung spielt eine große Rolle im Bereich der Fluktuation. Sie bezeichnet die Häufigkeit der Zu- und Abgänge im Projekt. Oft wird sie als Gütefunktion für die Führungskraft angesehen [Whita97] [deMar91]. In diese Betrachtung darf aber nicht nur eine niedrige Fluktuationsrate einfließen. Kontinuität in der Besetzung eines Projektes hat viele Vorteile, aber auch Nachteile. So werden einerseits durch wenig Wechsel im Projekt eingespielte Prozesse und enormes Wissen über das Projektumfeld, sowohl in technischer als auch sozialer Sicht, aufgebaut, andererseits kann eine andere Ansicht, durch einen neuen Kollegen eingebracht, zum Überdenken und eventuell zur Verbesserung eingefahrener Strukturen veranlassen [Barto97].

2.1.4 Beziehung zum Kunden

Ein wichtiger Aspekt in Softwareprojekten ist die Beziehung zu Beteiligten außerhalb des Projektes. Dies ist einerseits das eigene gehobene Management und andererseits der Kunde. Die richtige Beziehung zum Kunden kann helfen, viele Probleme bereits im Vorfeld zu umgehen. Kritische Arten der Beziehung sind eine zu geringe Einbeziehung des Kunden, ein schlechtes Verhältnis auf persönlicher Ebene, aber auch, für viele vielleicht erstaunlich, eine zu gute und enge Beziehung zum Kunden.

Zu geringe Einbeziehung des Kunden

Ein häufiger Fehler in Softwareprojekten ist die Meinung, alles besser zu wissen als der Kunde. Dies führt dazu, Entscheidungen ohne den Kunden zu treffen. Diese nicht abgesprochenen Entscheidungen erweisen sich im Nachhinein meistens als falsch [Weltz92]. Des Weiteren hat eine zu geringe Einbeziehung des Kunden in Entscheidungen sehr oft Mißverständnisse zur Folge. Unklarheiten werden nicht mit dem Kunden besprochen, sondern selbständig entschieden. Diese Fehler werden oft erst sehr spät, typischerweise bei einem Abnahmetest, gefunden, sind dann sehr schwierig zu korrigieren und daher sehr teuer [Ludew99]. Als Lösung dieser Probleme werden Kundenmeetings einberufen. Hierbei sollte es sich nicht nur um Review- oder Abnahmesitzungen handeln. Regelmäßige Kundenmeetings im Projektverlauf tragen zu einer besseren Akzeptanz und Transparenz des Gesamtprojektes bei.

Schlechte Beziehung zum Kunden

Einem schlechten Klima in der Beziehung zwischen den Entwicklern und dem Kunden liegen oft Streitigkeiten über den Vertrag zugrunde. Dies führt nicht selten dazu, daß Kunde und Hersteller nur noch schriftlich oder über höhere Management-Ebenen miteinander kommunizieren. Solche Schwierigkeiten führen zu o.g. geringer Einbeziehung des Kunden und nicht selten sogar zum Abbruch eines Projektes [Jones96].

Zu gute Beziehung zum Kunden

Nicht jede gute Beziehung zum Kunden oder häufige Abstimmung ist positiv für ein Projekt. Erstaunlicherweise existieren Beispiele, bei denen durch einen sehr engen, meist informellen Kontakt zwischen Entwicklern und Kunden ein weiteres Problem auftritt. Der enge Kontakt zum Kunden veranlaßt die Entwickler dazu, Wünsche des Kunden "mal eben schnell" umzusetzen, ohne offizielle Wege (wie z.Bsp. Change-Requests) einzuhalten. Dies führt dazu, daß sich die Entwickler mit Dingen befassen, die, zumindest zur Zeit, nicht zu ihren Aufgaben gehören und von denen andere nichts wissen. Hieraus resultieren oft Schwierigkeiten wie bsp. Terminverzögerungen, nicht abgesprochene Änderungen oder zusätzliche, nicht vertraglich vereinbarte Funktionen [Weltz92].

2.1.5 Beziehung zum Senior Management

Die Beziehung zwischen dem Projektleiter und dem gehobenen Management sollte bilateral verlaufen. Der Projektleiter muß das Projekt regelmäßig und vor allem wahrheitsgemäß dem gehobenen Management vorstellen. Zu wenige oder falsche Informationen führen zu Mißtrauen des Managements gegenüber dem Projekt und zu geringer Bereitschaft zu Eingeständnissen. Andererseits darf der Projektleiter nicht zu viele Eingeständnisse gegenüber dem gehobenen Management machen. Er muß das Projekt gegen unliebsame Einflüsse von außen, wie bspw. Budgetkürzungen oder Verschlechterung der Arbeitsbedingungen schützen [Ludew98]. Ein Projektleiter, der seine Mitarbeiter vor diesen Einflüssen schützt und sich vor sie stellt, erreicht zusätzlich einen enormen Motivations Schub seiner Mitarbeiter..

2.1.6 Teamzusammenstellung

Ein weiterer nicht zu vernachlässigender Aspekt im Verhalten von Mitarbeitern in Softwareprojekten ist das personelle Umfeld, anders gesagt das Team. Im Bereich der Teamzusammenstellung gilt es auf viele Einzelaspekte zu achten.

Art der Mitarbeiter

Jeder Mitarbeiter ist verschieden. So gelten EDV-Spezialisten als eine besondere Gattung von Mensch. Sie legen typisch keinen übermäßigen Wert auf soziale Kontakte und gelten als einfach zu führen [Couga78]. Es gibt aber nicht nur den typischen EDV-Spezialisten in Projekten. Heutzutage muß mit allen Varianten von Persönlichkeiten gerechnet werden. In der Literatur wird dies durch eine Einteilung in verschiedene Menschenbilder reflektiert. Die beiden wichtigsten werden im folgenden vorgestellt [Hentz95].

Menschenbild-Konzept nach MC GREGOR

McGregor unterscheidet zwischen Theorie X und Theorie Y :

Die Theorie X besagt, daß der Mensch grundsätzlich Arbeit ablehnt und versucht, ihr so weit wie möglich aus dem Weg zu gehen. Er muß dadurch ständigen Kontrollen unterworfen werden, da seine Ziele im Gegensatz zu den Zielen der Organisation stehen. Er strebt nach Sicherheit, hat keinerlei Ehrgeiz, sucht seine Befriedigung außerhalb der Arbeit und vermeidet Verantwortung. Er muß streng geführt werden.

Die Theorie Y geht davon aus, daß Arbeit für den Menschen einen natürlichen Bestandteil seines Lebens bildet. Er entwickelt Eigeninitiative und ist fähig zur selbständigen Steuerung und Kontrolle seines Verhaltens im Hinblick auf Ziele, die er akzeptiert und denen er sich verpflichtet fühlt. Die Erfüllung der Organisationsziele durch den Menschen ist abhängig von der dadurch möglichen Erfüllung seiner Individualziele. Er ist Träger des Kreativitäts- und Flexibilitätspotentials der Organisation und sucht Verantwortung.

Menschenbild-Konzept nach SCHEIN

Schein unterscheidet zwischen dem rational-ökonomischen, sozialen, sich selbst verwirklichenden und komplexen Mensch.

Der rational-ökonomische Mensch bezieht seine Motivation hauptsächlich aus monetären Anreizen, ist passiv, manipulierbar und durch rationale Maßnahmen zu steuern.

Der soziale Mensch bezieht seine Motivation aus kommunikativen und sozialen Beziehungen. Er zieht sich in einer mechanischen Arbeitsumgebung in soziale Beziehungen zurück und wird durch die Einbindung in die Gruppe stärker beeinflusst als durch Vorgesetzte. Er akzeptiert Führungshandlungen nur, wenn sie seine sozialen Bedürfnisse berücksichtigen.

Der sich selbst verwirklichende Mensch sieht Arbeit positiv, wenn sie der Selbstverwirklichung dient, die für ihn an oberster Stelle einer Bedürfnishierarchie steht. Er kann durch eigenverantwortliche und rationale Entscheidungen die Organisation unterstützen und sich selbst kontrollieren und motivieren.

Der komplexe Mensch ist lern- und wandlungsfähig. Er hat veränderliche Motive, die seine Sichtweise seiner Stellung in der Organisation beeinflussen und seine Entwicklungsbedürftigkeit befriedigen. Er hat in unterschiedlichen Situationen differenzierte Ziele.

Heute hat sich die Erkenntnis durchgesetzt, daß sich ein Mensch nicht in eine Schablone pressen läßt, sondern vielmehr eine Mischung aus diesen Menschenbildern darstellt. So sind zwar alle Ausprägungen vorhanden, bei den meisten Menschen ist jedoch eine Tendenz zu einem bestimmten Menschenbild erkennbar. Im Bereich der Softwareprojekte ist, wie bereits oben erwähnt, der soziale Mensch weniger anzutreffen. Studien zur Motivation von EDV-Spezialisten [Couga78] zeigen, daß es sich hierbei vor allem um sich selbstverwirklichende und rational-ökonomische Menschen handelt.

Natürlich müssen diese Menschenbilder in vielen Aspekten berücksichtigt werden. So haben die verschiedenen Bedürfnisse Auswirkungen auf das Teamwork und natürlich auf die Führung, die ein solcher Mensch akzeptiert. Je nachdem wie gut oder schlecht seine Bedürfnisse befriedigt werden, wird sich dies auf die Motivation des Mitarbeiters auswirken.

Teamwork

Gutes Teamwork ist einer der meist gewünschten Aspekte im Rahmen der Teambildung. Unter Teamwork versteht man die Fähigkeit, individuelle Stärken von verschiedenen Mitarbeitern zu einem hochproduktiven Portfolio zusammenzufassen. Um dies zu erreichen, müssen, wie bereits aus der Definition ersichtlich wird, Mitarbeiter mit verschiedenen, sich ergänzenden Fähigkeiten zusammenarbeiten. Diese Fähigkeiten dürfen nicht nur auf den technischen Bereich beschränkt sein, sondern müssen auch den sozialen Bereich erfassen [McDon99]. Das Finden solcher Teams wird meistens den Managern überlassen. Eine sehr interessante Alternative kann aber auch sein, den Teams selbst diese Entscheidung zu überlassen. So wurde beispielsweise ein Wettbewerb ausgeschrieben, bei dem sich selbst formierte Teams um bestimmte Projektaufgaben bewerben konnten. Dies führte zu hochproduktiven und hochmotivierten Teams [deMar91].

Durch die zunehmende Globalisierung, vor allem im Bereich der Datenverarbeitung, werden heutzutage internationale Teams, oft aus hochspezialisierten Experten, gebildet. Hierdurch entstehen neue Risikofaktoren. So muß außer den üblichen Schwierigkeiten noch auf kulturelle, religiöse oder politische Aspekte Rücksicht genommen werden. Es sollte beispielsweise eine Einladung zum Geschäftsessen mit dem islamischen Testexperten während des Ramadans zumindest überdacht werden.

Schlüsselpositionen

In verschiedenen Teams hat sich immer wieder gezeigt, daß es in den meisten Fällen Mitarbeiter gibt, die eine Schlüsselposition innehaben [Weltz92]. Mit dieser Person steht und fällt oft das gesamte Team, in positiver und negativer Hinsicht. Entsprechend groß sind die Auswirkungen auf das Projekt, wenn einer dieser Mitarbeiter aus dem Projekt ausscheidet. Die Variation der Schlüsselpositionen reicht vom Chief-Programmer [Ludew99] über den informellen "Kommunikationskatalysator", jemandem, der zu einer besseren Kommunikation unter den Mitarbeiter beiträgt [deMar91], bis hin zu fast intriganten Mitarbeitern [Boddi99]. Im Erkennen von und im Umgang mit solchen Mitarbeitern liegt einer der Schlüsselfaktoren, um erfolgreiche Teams zu managen.

2.1.7 Mitarbeiterführung

Auch die Mitarbeiterführung ist sicherlich einer der Schlüsselfaktoren und damit auch einer der größten Krisenherde, in einem Softwareprojekt [Thaye97]. Vom Führungsgeschick hängt es oftmals ab, wie erfolgreich oder erfolglos ein Projekt abgeschlossen wird. Natürlich läßt sich der Aspekt der Mitarbeiterführung nicht in einen kurzen Abschnitt fassen, sondern könnte ganze Bücher füllen (und tut dies auch). In diesem Dokument werden nur zwei Unter Aspekte vorgestellt, die im Rahmen eines Softwareprojektes der Literatur zufolge den größten Einfluß haben. Erstens ist dies der Führungsstil des Projektleiters und zweitens die Belastung der Mitarbeiter.

Führungsstil

Über Führungsstile existieren lange Abhandlungen. In diesem Kapitel wird kurz der aktuelle Stand der Forschung zum Thema Führungsstile und deren Anwendung wiedergegeben.

Bereits früh wurden einige traditionelle Führungsstile identifiziert, die in der Praxis fast ausschließlich auftreten. Diese Führungsstile werden heute als klassische Führungsstile bezeichnet [Staehe87 nach Maslo54]. Schon bald wurde aber klar, daß sich Personalführung nicht in einheitliche Schemen pressen läßt. Aus dieser Erkenntnis heraus entstand die Kontinuum-Theorie, die einen Führungsstil als den Faktor der Mitbestimmung der Geführten bezeichnet [Tanne58]. Die Führungsmöglichkeiten wurden auf einer Skala angeordnet und in sechs Bereiche eingeteilt:

Autoritärer Führungsstil:

Der Vorgesetzte entscheidet allein.

Patriarchalischer Führungsstil:

Der Vorgesetzte entscheidet, versucht aber seine Mitarbeiter von seiner Entscheidung zu überzeugen.

Beratender Führungsstil:

Der Vorgesetzte entscheidet und ist bestrebt, entsprechende Fragen der Mitarbeiter zu beantworten, um dadurch die Akzeptanz zu erreichen.

Kooperativer Führungsstil:

Der Vorgesetzte berücksichtigt die Meinungen der Mitarbeiter, bevor er die endgültige Entscheidung trifft.

Partizipativer Führungsstil:

Die Mitarbeiter entwickeln gemeinsam Problemlösungen und der Vorgesetzte entscheidet sich für die von ihm Favorisierte.

Demokratischer Führungsstil:

Entweder entscheidet die Gruppe, nachdem Grenzen und Entscheidungsspielraum von dem Vorgesetzten erläutert wurden, oder der Vorgesetzte agiert als Koordinator nach innen und nach außen.

Die Aufzählung beginnt mit dem Führungsstil, bei dem der Mitarbeiter am wenigsten und der Vorgesetzte am meisten Entscheidungsspielraum hat (autoritärer Führungsstil). Bis hin

zum demokratischen Führungsstil nimmt der Entscheidungsspielraum des Mitarbeiters zu und der des Vorgesetzten ab (demokratischer Führungsstil).

Eine Forderung aus der Praxis ist seit jeher, einen Bausatz zu bekommen, anhand dessen man sich den für sein Team passenden Führungsstil auswählen kann. Aus diesem Grund wurde in der Forschung in vielen Arbeiten versucht, einen optimalen Führungsstil zu erarbeiten. Allerdings wurde in den vielen Versuchen bis heute keine einfache, allgemeingültige Lösung erreicht. Neueste Lösungen beruhen auf dem Situationsansatz [Hentz95]. In diesem Ansatz wird keine eindeutige Lösung präsentiert, sondern es werden die Kriterien genannt, die zur Auswahl eines Führungsstiles herangezogen werden müssen. Hierbei handelt es sich einerseits um die Eigenschaften des Führenden (Bspw.: Erfahrung, Autorität, Sachkenntnis) und der Geführten (Bspw.: Ausbildungsstand, Erfahrung, Eigenständigkeit), andererseits um die aktuelle Situation, in der sich der Führende und die Geführten befinden (Bspw.: Unterstützung durch das Management, extreme Streßsituation, Art der Aufgabe). Anhand dieser Bewertungen werden dann Empfehlungen abgegeben, in welchen Bereichen der Skala der Kontinuum-Theorie von Tannenbaum man sich bewegen sollte. Für die heutige Zeit werden eher demokratische Führungsstile empfohlen. Besonders wenn die Angestellten eine hochwertige Ausbildung besitzen und eigenständiges Arbeiten gewohnt sind. Aufgrund dieser zwei Faktoren läßt sich also für ein Softwareprojekt ein Führungsstil empfehlen, der starke demokratische Bestandteile enthält.

Heute werden in der Praxis oft Führungstechniken empfohlen und eingesetzt, die mit dem modernen Namen "Management by..." beginnen. Die am weitesten verbreitete und wohl auch erfolgreichste ist das "Management by Objectives" (Führung durch Zielorientierung) [Hentz95]. Hierbei werden durch den Manager und den Geführten gemeinsam Ziele vereinbart, die der Mitarbeiter innerhalb eines bestimmten Zeitraumes, oft ein Jahr, erreichen möchte. Diese Ziele müssen natürlich in Einklang mit den Projekt-, Abteilungs- und Unternehmenszielen stehen. Anhand von Soll-/Ist-Vergleichen läßt sich eine Leistungskontrolle und -beurteilung einfach vornehmen.

Über- oder Unterforderung

Ein weiterer Punkt aus Sicht der Mitarbeiter ist die Arbeitsbelastung, die in einem Projekt auftritt. So führen übermäßiger Streß und stetige Überstunden zu einer Überforderung und schließlich zum "Burn-Out", einer von Resignation geprägten Phase von Produktivitätsverlust, die meist von Krankheiten oder persönlichen Problemen begleitet wird [Weltz92] [deMar91]. Andererseits ist auch eine ständige Unterforderung dem Projekt nicht zuträglich. Mitarbeiter, die ständig unterfordert sind, verlieren schnell die Motivation und verlassen dann das Projekt. Allgemein wird eine leichte Überlast an Arbeit als optimal für den Projektverlauf und die Produktivität angesehen [Elzer89]. Es ist sicherlich nicht einfach, das richtige Maß an Forderung zu erreichen. Mitarbeitergespräche sind ein gutes Mittel, um relativ sicher feststellen zu können, ob der Mitarbeiter im richtigen Maß gefordert ist.

2.2 Betrachtung der Einflüsse und Auswirkungen

In diesem Unterkapitel werden die in der Aspektdefinition erarbeiteten Aspekte in Beziehung zueinander und zu anderen Aspekten in Softwareprojekten gesetzt. Es werden

sowohl die Einflüsse auf diese Aspekte als auch ihre Auswirkungen im Projekt beschrieben. Diese Zusammenhänge werden größtenteils aus der Literatur abgeleitet. Sollten für einige Aspekte keine plausiblen oder ausreichenden Zusammenhänge erkennbar sein, werden diese durch Hypothesen ergänzt und entsprechend gekennzeichnet. Ziel dieses Abschnittes ist es, einen Überblick über diese Aspekte in Softwareprojekten im Zusammenhang mit Verhaltensaspekten generell zu schaffen. In diesem Kapitel erfolgt noch keine Anpassung an die speziellen Gegebenheiten des SESAM-Projektes, bis auf die Reduzierung auf kleine Auftragsprojekte und den Ausschluß von einigen Zusammenhängen, die im Rahmen des SESAM-Simulators keinesfalls realisierbar sein werden.

Um im späteren Verlauf der Arbeit eine Systematik zur Umsetzung der identifizierten Aspekte in das QS-Modell aufstellen zu können, wurden die Zusammenhänge in diesem Kapitel bereits grob in Inputs und Outputs aufgeteilt. Inputs bezeichnen hierbei alle Einflüsse und Aspekte, die sich auf den beschriebenen Aspekt auswirken und ihn beeinflussen. Als Outputs werden die Effekte bezeichnet, durch die der definierte Aspekt Auswirkungen auf andere Aspekte hat und diese beeinflusst.

2.2.1 Motivation

Der Aspekt der Motivation hat sich im Verlauf der Arbeit immer mehr als zentraler Aspekt im Zusammenhang mit menschlichem Verhalten herausgestellt. So baut diese Arbeit in vielen Aspekten auf den Grundlagen der Motivationstheorie auf. Um den Gesamtzusammenhang der Arbeit erfassen und die im weiteren Verlauf der Arbeit getroffenen Entscheidungen nachvollziehen zu können, werden die Einflüsse und Auswirkungen der Motivation in diesem Kapitel sehr ausführlich erklärt.

Entstehung von Motivation

Schon Maslow [Maslo54] erkannte frühzeitig, daß Motivation auf der Befriedigung von menschlichen Bedürfnissen beruht. Er definierte fünf grundlegende Bedürfnisse, mit denen er die Motivation beschrieb. Einen Schritt weiter ging Herzberg [Herzb59] in einer empirischen Studie. Die daraus abgeleitete Dual-Faktoren-Theorie soll hier, stark vereinfacht und zusammengefaßt, kurz wiedergegeben werden. Herzberg erkannte, daß sich die (Arbeits-) Motivation eines Menschen aus verschiedenen Faktoren zusammensetzt. Diese Faktoren lassen sich in zwei Gruppen aufteilen. Einerseits existieren Zufriedenmacher, die die Motivation eines Mitarbeiters in der richtigen Ausprägung ansteigen lassen. Sind diese Faktoren nicht in der gewünschten Ausprägung vorhanden, sinkt die Motivation jedoch nicht oder nur geringfügig. Die andere Gruppe der Faktoren sind die Hygienefaktoren. Sind sie in der richtigen Ausprägung vorhanden, steigt die Motivation nicht oder nur gering und kurzfristig an. Sind diese Faktoren jedoch nicht in der richtigen Ausprägung vorhanden, hat dies einen starken Motivationsabfall zur Folge. Zufriedenmacher wirken sich also fast ausschließlich motivationssteigernd aus, während Hygienefaktoren fast ausschließlich motivationssenkende Auswirkungen haben.

Herzberg identifizierte 16 Faktoren, die er wie folgt in die beiden Gruppen einteilte:

Zufriedenmacher (Motivatoren)	Unzufriedenmacher (Hygienefaktoren)
Leistungserlebnis	Unternehmenspolitik
Anerkennung	Führungstechniken
Arbeitsinhalt	Entlohnung
Verantwortung	Beziehung zum Vorgesetzten
berufl. Aufstieg	Arbeitsbedingungen
Entwicklungsmöglichkeiten	Beziehung zu Arbeitskollegen
	Privatleben
	Status
	Beziehung zu Untergebenen
	Arbeitsplatzsicherheit

*Tabelle 1: Arbeitsmotivationsfaktoren nach Herzberg [Herzb59]
(nach Wichtigkeit sortiert)*

Die einzelnen Faktoren lassen sich wie folgt beschreiben:

Leistungserlebnis

Eine sehr starke Motivation bezieht ein Mitarbeiter aus der Tatsache, daß seine Arbeit ein gewisses Resultat liefert. Ist dieses Resultat seinen Erwartungen entsprechend, steigt die Motivation an. Eine Leistung, die über den eigenen Erwartungen liegt oder mit der man persönlich sehr zufrieden ist, zieht, wie es sicherlich jeder bereits erlebt hat, einen gewaltigen Motivationsschub nach sich. Oft entstehen in diesem Fall Gedanken wie: "Wenn ich das schaffe, dann kann ich alles schaffen". Dagegen wird eine Arbeit, die kein entsprechendes Ergebnis liefert, oft als sinnlos angesehen. Man arbeitet nicht gerne "für den Papierkorb". Diese Tatsache führt aber meistens nicht zu einem starken Motivationsabfall, da man sich relativ schnell an eine bezüglich des Ergebnisses nicht zufriedenstellende Tätigkeit gewöhnt und dann resigniert vor sich hinarbeitet.

Anerkennung

Der Faktor "Anerkennung" tritt oft ergänzend zum Faktor "Leistungserlebnis" auf. Durch Anerkennung wird der Mitarbeiter darauf aufmerksam gemacht oder darin bestätigt, daß seine Ergebnisse eine gute Leistung darstellen.

Arbeitsinhalt

Der Inhalt der Arbeit stellt für einen Mitarbeiter einen hohen Motivationsfaktor dar. Hieran läßt sich gut das Prinzip eines Motivators erläutern. Eine interessante Arbeit kann die Motivation in einem hohen Maß steigern, so daß hierfür auch einige fehlende Hygienefaktoren in Kauf genommen werden. So kann beispielsweise ein geringeres Gehalt durchaus durch eine sehr interessante Tätigkeit ausgeglichen werden. Im negativen Fall ist diese Auswirkung jedoch nicht so stark. Eine uninteressante Tätigkeit senkt die Motivation

nicht in großem Ausmaß. Man findet sich mit dieser Tätigkeit ab, erfährt dadurch zwar keine Motivationssteigerung aber auch eine tiefe Depression bzw. starke Motivationsenkung findet nicht statt..

Verantwortung

Der Faktor "Verantwortung" verhält sich analog zum Faktor "Arbeitsinhalt". Eine Tätigkeit mit hoher Verantwortung führt zu einer höheren Motivation. Ist die Tätigkeit nicht sehr verantwortungsvoll, so trägt dies aber nicht zur Motivationsminderung bei. Die gleiche Tätigkeit ist nicht demotivierender ohne zusätzliche Verantwortung. Sie kann trotzdem interessant und ausfüllend sein.

Beruflicher Aufstieg

Der berufliche Aufstieg zeigt auch die klaren Merkmale eines Motivators. Eine Beförderung wirkt sehr motivierend, da man wiederum auf das Leistungserlebnis einen Bezug nehmen kann. Eine Beförderung gibt einem das Gefühl, etwas richtig gemacht zu haben und einer der Besten in seinem Job zu sein. Eine Tätigkeit kann jedoch, wie bei den anderen Motivatoren, auch ohne Beförderung interessant sein und senkt deshalb nicht die Motivation. Anders kann dies im Fall einer versprochenen, aber nicht durchgeführten Beförderung oder bei einer übergangenen Beförderung sein. Hier kann dieser Faktor sich in's Negative umwandeln. Herzberg sieht diese Auswirkungen aber als Fehlverhalten des Führenden bzw. des Unternehmens an und ordnet diese in das Gebiet "Führungsverhalten" bzw. "Unternehmenspolitik" ein (s.u.).

Entwicklungsmöglichkeiten.

Existieren in einem Beruf gute Weiterentwicklungsmöglichkeiten und werden diese auch angeboten und durchgeführt, so kann dies die Motivation steigern. Auch hier läßt sich ein Bezug zum Leistungserlebnis bzw. zum zu erwartenden Leistungserlebnis schaffen. Der Mitarbeiter ist der Meinung, durch eine bessere Ausbildung noch mehr leisten zu können, damit dann auch verantwortungsvollere Aufgaben zu bekommen und ein noch höheres Leistungserlebnis zu haben. Nicht vorhandene Entwicklungsmöglichkeiten senken ebenfalls die Motivation nicht. Ein interessanter Job ist auch ohne Weiterentwicklung interessant. Jedoch kann es bei einer sehr langen Phase ohne Entwicklungsmöglichkeiten schnell zu einer Monotonie kommen. Dies schlägt sich dann negativ auf den Punkt "Arbeitsinhalt" nieder.

Unternehmenspolitik / Arbeitsbedingungen

Die Unternehmenspolitik und die Arbeitsbedingungen liefern kaum Potential um die Motivation zu verbessern. Unternehmenspolitik ist für viele Mitarbeiter nur die Anzahl der "Knüppel", die man zwischen die Beine geworfen bekommt. Ist diese Anzahl zu hoch, dann wirkt sie sich enorm demotivierend aus.

Führungstechniken / Beziehung zum Vorgesetzten

Dieser Punkt wurde an Herzbergs These oft diskutiert. Unzählige Managementbücher propagieren die Führungstechnik als das Mittel zur Mitarbeitermotivation, während Herzberg sie nur als Hygienefaktor ansieht, also nur als Mittel, um Schlimmeres zu vermeiden. Betrachtet man aber die Mehrzahl der Angestellten, so wird man kaum einen finden, der

zur Arbeit geht weil der Chef so nett ist. Andererseits kündigen jedes Jahr zahllose Mitarbeiter aufgrund von Führungsfehlverhalten der Führungskräfte.

Entlohnung

Der Faktor Entlohnung wird am häufigsten in Herzbergs Theorie diskutiert. Viele Kritiker sind der Meinung, die Entlohnung wäre vielmehr ein Motivator als ein Hygienefaktor. Herzberg begründet seine Eingruppierung damit, daß eine höhere Entlohnung nur kurzfristige Motivationssteigerung bringt, vor allem aber im Fall einer nicht angemessenen Entlohnung schweren Motivationsverlust nach sich zieht.

Privatleben

Dies ist ein Punkt, der oft wenig bedacht wird. Auch Probleme im Privatleben eines Mitarbeiters können sich negativ auf die Motivation auswirken. Der Mitarbeiter hat einfach anderes im Kopf als seine Arbeit, da das Privatleben als wichtiger angesehen wird. Ein besonders harmonisches Privatleben steigert jedoch nicht die Motivation, sondern wird als normal angesehen.

Beziehung zu Arbeitskollegen / Status

Die Beziehung zu Arbeitskollegen und der Status innerhalb einer Gruppe verhalten sich da etwas anders. Es gibt durchaus Menschen, die gern zur Arbeit gehen, weil sie in einem guten Team arbeiten, sich mit den Kollegen gut verstehen und angesehen sind. Jedoch wird die Motivation dadurch nicht in dem Maße gesteigert, wie sie sich senken würde, wenn man mit den Kollegen nicht gut auskommt.

Beziehung zu Untergebenen

Dieser Punkt trifft vor allem für Führungskräfte zu. Die Führungskraft erwartet von sich selbst oft, eine gute Beziehung zu Untergebenen zu haben. Ist dies nicht der Fall und trifft man nur auf Konfrontationen, so ist dies enorm motivationsenkend.

Arbeitsplatzsicherheit

Auch die Arbeitsplatzsicherheit wird als selbstverständlich angesehen. Dieser Faktor kommt meist nur dann ins Spiel, wenn man in Gefahr kommt, seinen Arbeitsplatz zu verlieren. In diesem Fall bringt eine große Angst den Mitarbeiter dazu, vorsichtig zu agieren, nichts falsch zu machen und alles genauestens zu kontrollieren, um Fehler zu vermeiden. Diese Angst und eine andere Arbeitseinstellung führen zu einem Motivationsverlust mit allen daraus resultierenden Folgen.

All diese Faktoren sind jedoch nicht bei allen Menschen gleich ausgebildet und gleich gewichtet. Jeder legt für sich eine persönliche Rangfolge unter allen Faktoren fest. Oft stimmen diese Rangfolgen innerhalb einzelner Menschengruppen (siehe Ausprägungen von Menschenbildern) erstaunlich eng überein. Im Kapitel 2.2.6 werden diese Faktoren für den EDV-Spezialisten gewichtet.

Auswirkungen von Motivation

Herzberg untersuchte des weiteren die Auswirkungen eines Erlebnisses, das positive oder negative Folgen auf die Motivation hat. Hierbei erkannte er fünf Hauptfolgen, die sowohl bei positiven als auch bei negativen Erlebnissen auftraten.

Auswirkungen besonderer Zufriedenheit	Auswirkungen besonderer Unzufriedenheit
Leistung (quantitativ und qualitativ)	Einstellungsänderungen bzgl. Mitmenschen, Firma, Beruf, Vertrauensbasis
Einstellungsänderungen bzgl. Mitmenschen, Firma, Beruf, Vertrauensbasis	Leistung (quantitativ und qualitativ)
Zwischenmenschliche Beziehungen	Gesundheit (Streßauswirkungen, psychosomatische Störungen)
Gesundheit	Zwischenmenschliche Beziehungen
Fluktuation (Unterlassung einer Kündigung)	Fluktuation (innere oder äußere Kündigung)

Tabelle 2: Auswirkungen besonderer Zufriedenheit bzw. besonderer Unzufriedenheit nach Herzberg (nach Auftretenshäufigkeit geordnet)

Leistung

Dies ist die insgesamt wohl am häufigsten festgestellte Auswirkung von Motivation. Die Produktivität eines Mitarbeiters ist sehr eng mit dessen Motivation verbunden [Couga78]. Ein motivierter Mitarbeiter fühlt sich an seinem Arbeitsplatz wohl und ist mit seinen Leistungen zufrieden. Er ist der Meinung, daß er auch mehr schaffen könnte. Dies versucht er nun zu erreichen.

Einstellungsänderungen / Zwischenmenschliche Beziehungen

Auswirkungen auf die Einstellung und die zwischenmenschlichen Beziehungen haben wiederum Auswirkungen auf die Motivation selbst. Ein negatives Erlebnis in Bezug auf die Beziehung zu den Arbeitskollegen senkt die Gesamtmotivation. Ist diese niedrig, so wird hierdurch wiederum die Beziehung zu den Arbeitskollegen verschlechtert, da niemand gerne mit ständig unmotivierten Mitarbeitern verkehren möchte. Es entsteht also ein gefährlicher Teufelskreis, der durch die Führungskraft gestoppt werden muß, da es sonst früher oder später durch die folgende Aufschaukelung zu einer starken Senkung der Gesamtmotivation im Team und damit zu anderen Folgen wie bspw. einer hohen Fluktuationsrate kommen wird.

Einstellungsänderungen verschlimmern außerdem oft negative Erlebnisse und verzögern ihr Auftreten ("man frißt den Ärger in sich hinein"). Dies ist also ein Punkt der durch eine

Führungskraft sehr genau beobachtet und an dem rechtzeitig eingegriffen werden muß um schlimmeres zu verhindern.

Gesundheit

Die Gesundheit, beziehungsweise die Häufigkeit der Fehlzeiten, ist ein Faktor, der nach außen hin die Motivation eines Mitarbeiters einigermaßen sichtbar macht. Schon seit längerer Zeit ist bekannt, daß eine geringe Motivation zu einem hohen Krankenstand führt. Dies soll nicht heißen, daß ein Mitarbeiter, der drei Monate im Krankenhaus liegt, eine schlechte Motivation besitzt. Nicht alle Krankheiten hängen von der Motivation ab. Viele einzelne, verstreute Krankentage sollten einer Führungskraft aber doch zu denken geben. Hier wäre es eventuell angebracht, mir Hilfe von Mitarbeitergesprächen genauer nachzuforschen.

Fluktuation

Die am meisten gefürchtete Folge von geringer Motivation ist die Kündigung. Während ein positives Erlebnis einen davon abbringen kann zu kündigen, ist es bei negativen Erlebnissen oft der letzte Lösungsweg.

Der eigentlichen Kündigung geht in den meisten Fällen eine innere Kündigung voraus. Diese baut sich in mehreren Schritten auf. Es beginnt oft damit, daß sich der Mitarbeiter Stellenanzeigen in Tageszeitungen durchliest und kann sich letztendlich über aktive Bewerbungen bei anderen Firmen bis hin zur Resignation im eigenen Beruf steigern. Diesen Zustand bezeichnet man als innere Kündigung. Dies kann sogar soweit führen, daß man entweder Dienst nach Vorschrift betreibt oder versucht, kontraproduktiv zu arbeiten. Oft kommt es aber nicht bis zu diesen Schritten, sondern es erfolgt rechtzeitig eine äußere Kündigung, nämlich der Wechsel in eine andere Firma oder Abteilung.

2.2.2 Produktivität

Die Produktivität ist von einer Vielzahl von Aspekten abhängig. Als Beispiele seien hier die Erfahrung der Mitarbeiter, die Teamgröße und die Art der Aufgabe genannt. Da die Produktivität bereits im QS-Modell integriert ist [Drapp00], wird an dieser Stelle keine genaue Erörterung der Zusammenhänge mehr vorgenommen. Die Produktivität wird nur durch eine Abhängigkeit von der Motivation ergänzt (siehe Kap. 2.2.1).

2.2.3 Fluktuation

Die Fluktuation hat erwiesenermaßen großen Einfluß auf ein Softwareprojekt. Wie bereits in der Aspektdefinition (Kap. 2.1.3) wird die Fluktuation in Abgänge, Zugänge und die Kontinuität der Besetzung unterteilt.

Abgänge

Die Auswirkungen des Ausscheidens von Mitarbeitern sind oft und häufig beleuchtet worden. Während Metzger [Metzg96] nur feststellt, daß man "ein Problem hat", wenn Mitarbeiter aus dem Projekt ausscheiden, erläutert Bartol [Barto97], daß ein Mitarbeiterabgang sowohl positive als auch negative Folgen haben kann. Als negativ sieht er folgende Auswirkungen an:

Verlust von undokumentierten Informationen und von Hintergrundwissen

In keinem Projekt sind wirklich alle Informationen festgehalten. Durch den Abgang eines Mitarbeiters gehen immer nicht dokumentierte Informationen und vor allem Erfahrungen

im Projektumfeld verloren. Diese Informationen müssen durch andere Projektmitglieder erneut, also doppelt, erarbeitet und erfaßt werden. Je länger ein Mitarbeiter im Projekt war, desto mehr dieser Informationen und Erfahrungen sind normalerweise vorhanden und desto schwerer wiegt deren Verlust.

Durch die fehlende Arbeitskraft kommt es, zumindest kurzfristig, zu einer Mehrlast an Arbeit für die restlichen Mitarbeiter.

Das Arbeitsvermögen des ausgeschiedenen Mitarbeiters kann in den allermeisten Fällen nicht sofort ersetzt werden. Die übriggebliebene Arbeit muß deshalb von den anderen Mitarbeitern miterledigt werden. Dies führt aufgrund der Mehrbelastung, eventueller Überstunden und der für die anderen Mitarbeiter vielleicht nicht interessanten und ausfüllenden Tätigkeiten, zu einem Motivationsverlust, der zumindest temporär ist. Selbst wenn neue Mitarbeiter zum Projekt stoßen, bringt dies nicht direkt die erhoffte Erleichterung. Jeder neue Mitarbeiter muß sich zumindest kurzzeitig in das Projekt einarbeiten und erreicht in dieser Phase nicht die optimale Produktivität und bindet zusätzlich Ressourcen der Mitarbeiter, die ihn einarbeiten.

Vertretungen müssen erst eingearbeitet werden und verursachen dadurch höhere Kosten, Zeitverzögerungen und höhere Fehlerraten

Durch den Einsatz von Vertretungen bei temporärem Ausscheiden entstehen enorme Kosten und durch die fehlende Erfahrung (s.o.) auch Zeitverzögerungen. Je nachdem, wie gut oder schlecht eine Vertretung ist, sind diese Kosten und Verzögerungen unterschiedlich groß.

Sind bestimmte Fähigkeiten nicht verfügbar und auch nicht beschaffbar, kann es zum Abbruch des gesamten Projektes kommen.

Gerade in der heutigen Zeit, in der EDV-Fachkräfte sehr schwer zu beschaffen sind, ist es nicht immer möglich, einen bestimmten Mitarbeiter nach dessen Ausscheiden überhaupt zu ersetzen. Dies kann dazu führen, daß ein gesamtes Projekt nicht erfolgreich abgeschlossen werden kann.

Motivationsverlust der übrigen Mitarbeiter

Durch das Ausscheiden eines Mitarbeiters wird normalerweise zumindest für eine bestimmte Zeit die Motivation der anderen Mitarbeiter gesenkt. Dies entsteht dadurch, daß die anderen Mitarbeiter oft durch die Kündigung auf die Probleme und/oder Alternativen, die der Kündigende hatte, aufmerksam gemacht werden und diese auch für sich selbst entdecken.

Verschlechterung des Teamworks (auch in [Weltz92])

Verläßt ein beliebter Mitarbeiter das Team, dann wirkt sich dies auf die Teamzugehörigkeit aus. Hier zeigt sich eine Auswirkung auf das Verhältnis zu Arbeitskollegen und damit auf einen Hygienefaktor.

Das Ausscheiden eines Mitarbeiters wird überwiegend als negativ angesehen. Jedoch kann es auch einige positive Folgen nach sich ziehen, wenn ein Mitarbeiter aus einem Projekt ausscheidet. Diese Faktoren werden in den nächsten Abschnitten kurz erläutert:

Höhere Motivation der übrigen Mitarbeiter

Verläßt ein Mitarbeiter das Projekt, der bisher nur wenig oder eine schlechte Leistung gebracht hat, so gibt dies den restlichen Mitarbeitern das Vertrauen in das Projekt zurück. Hat ein unproduktiver “Mitläufer” das Projekt verlassen, bewirkt dies ein potentiell höheres Leistungserlebnis und ein besseres Verhältnis zu den Arbeitskollegen. Dies steigert nach Herzberg die Motivation und wird von Davis [Davis97] sogar als Operation (“Surgery”) bezeichnet, die schlechtes Gewebe herausschneidet.

Kostensparen durch geringe Anfangsgehälter

Durch das Ausscheiden von erfahrenen Mitarbeitern können neue Mitarbeiter eingestellt werden, die mit geringeren Anfangsgehältern in die Firma einsteigen. Hierdurch läßt sich eine enorme Kostenersparnis erzielen. Dieser Punkt könnte dazu verleiten, nur noch neue Mitarbeiter einzustellen. Die Erfahrung zeigt jedoch, daß der Verlust an Erfahrung und der Einarbeitungsaufwand meist schwerer wiegen als die Kostenersparnis.

Hinzufügen von “frischem Blut”

Neue Erfahrungen, Ansichten und Ideen die durch neue Mitarbeiter in das Projekt hineingebracht werden, können dazu führen, eingefahrene Prozesse zu überdenken. Dies hilft oft dabei, diese Prozesse zu verbessern und damit das Projekt intern schneller und effizienter zu gestalten. Es kann zwei Auswirkungen haben. Einerseits werden Kosten und Zeit eingespart und die Motivation der Mitarbeiter wird erhöht, da sich die Arbeitsbedingungen verbessern. Andererseits fühlen sich Mitarbeiter aber oft auch etwas “auf den Schlips getreten”, wenn ein neuer Mitarbeiter in das Projekt kommt und die alteingesessenen Prozesse anzweifelt und verbessern möchte. Dieser Motivationsverlust ist aber nur temporär, da die verbesserten Prozesse schon bald Wirkung zeigen und dann auch die vorher zweifelnden Mitarbeiter überzeugen.

Bessere Aufstiegschancen

Wenn höhergestellte Mitarbeiter aus dem Projekt aussteigen, ergibt sich für andere Mitarbeiter die Möglichkeit, in deren Positionen aufzusteigen. Durch einen Ausstieg werden also die Motivationsfaktoren beruflicher Aufstieg und Entwicklungsmöglichkeiten beeinflusst.

Zusammenfassend kann gesagt werden, daß sich ein Ausscheiden aus einem Projekt hauptsächlich negativ auf Zeitbedarf, Kosten und Motivation der übrigen Mitarbeiter auswirkt. In einigen seltenen Fällen, wenn ein ausscheidender Mitarbeiter nicht sehr beliebt oder unproduktiv war, überwiegen aber die positiven Auswirkungen.

Die Gründe für eine Kündigung stellen sich über alle Autoren hinweg sehr einheitlich dar. Cougar reduziert es auf die einfache These: Wenn die Motivation, die Jobzufriedenheit und die Qualität der Arbeit hoch ist, dann ist das Fehlen und die Kündigung gering.

Chandor [Chand71] gibt folgende Gründe für eine Kündigung an:

- Unzufriedenheit mit dem Gehalt
- Langeweile
- Ortswechsel
- Projektende

- Schlechte Arbeitsbedingungen
- Keine Anerkennung
- Keine Ausstiegsmöglichkeiten
- Kein Vertrauen in das Management

Alle diese Gründe lassen sich fast völlig auf die These von Herzberg (siehe Kapitel 2.2.1) abbilden, der diese Punkte als Gründe für eine schlechte Motivation ansieht, die wiederum eine Kündigung nach sich ziehen kann. Abschließend kann also gesagt werden, daß eine Kündigung vor allem durch eine geringe Motivation entsteht.

Zugänge

Die Auswirkungen von Zugängen wurden bereits teilweise als positive Punkte unter den Abgängen erwähnt und sollen hier nicht mehr wiederholt werden. Zusätzlich sind nur noch die Auswirkungen von Brooks [Brook74] zu erwähnen, der eine Senkung der Produktivität beim Hinzufügen von neuen Mitarbeiter voraussagt. Diese Punkte sind jedoch bereits in SESAM implementiert und werden hier deshalb nicht mehr ausführlich besprochen.

Kontinuität

Auch die Kontinuität der Besetzung hat positive und negative Auswirkungen.

Hypothese

Eine hohe Kontinuität fördert die Teamzusammengehörigkeit, damit das Teamwork, und steigert die Erfahrung der einzelnen Mitarbeiter. Außerdem wird durch eine Kontinuität in der Besetzung eine bessere Beziehung zum Kunden möglich. Man stellt sich mit der Zeit aufeinander ein.

Eine hohe Kontinuität kann auch negative Folgen haben. Durch eine sehr starke Teamzusammengehörigkeit bilden sich, wie oben erwähnt, oft feste Strukturen und eingefahrene Prozesse. Stößt nun nach langer Zeit einmal wieder ein neuer Mitarbeiter hinzu, der diese Strukturen und Prozesse, eventuell berechtigt, anzweifelt, so wirkt sich dies negativ auf die Motivation der "alteingesessenen" Mitarbeiter aus, die ihr Werk angezweifelt sehen.

Es kann also zusammenfassend gesagt werden, daß sich Kontinuität in einem gewissen Maß als positiv auf die Produktivität, die Beziehung zum Kunden, das Teamwork und die Projekterfahrung der Mitarbeiter auswirkt. Eine zu hohe Kontinuität, also keine Fluktuation, fördert aber auch extrem die Betriebsblindheit und damit die Verschlossenheit gegenüber Neuerungen und Verbesserungen. Nach einer sehr langen Zeit der Zusammenarbeit können auch negative Auswirkungen auf das Teamwork und die Beziehung zu den Arbeitskollegen auftreten. Man nennt dies in Sportlerkreisen z.B. "Lagerkoller".

2.2.4 Beziehung zum Kunden

Die Beziehung zum Kunden kann sich auf verschiedenste Weise auf ein Softwareprojekt auswirken. Wird der Kunde nur selten in das Projekt miteinbezogen, so äußert sich dies oft in Anforderungsänderungen, die erst sehr spät an das Projekt herangetragen werden [Weltz92]. Hierdurch ergeben sich extreme Zeitverzögerungen und eine höhere Arbeitsbelastung der Mitarbeiter, die in späten Phasen des Projektes Arbeiten erledigen müssen, die nicht eingeplant waren und eigentlich schon erheblich früher hätten durchgeführt werden

müssen. Hinzu kommt das hierdurch nun auch alle Ergebnisse früherer Phasen überarbeitet werden müssen, da auch dort die Fehler vorhanden sind.

Eine zu geringe Beteiligung des Kunden schlägt sich auch in einem geringen Verständnis füreinander wieder. Daraus folgt eine schlechte und unproduktive Beziehung zum Kunden mit vielen daraus resultierenden Konflikten, die gelöst werden müssen. [Jones96]. Dieses schlechte Verhältnis zum Kunden wirkt sich auch auf die Motivation der Mitarbeiter aus. Hier sind vor allem die Faktoren Arbeitsbedingungen und Leistungserlebnis zu nennen. Durch die schlechte Beziehung zum Kunden werden die auch Arbeitsbedingungen durch eine aufwendigere und anstrengendere Kommunikation mit dem Kunden schlechter und das Leistungserlebnis wird durch einen Kunden, der ständig negative Kritik vorbringt, sehr gemindert.

Hypothese

Durch eine zu geringe Einbeziehung des Kunden sinkt auch die Motivation des Kunden, da das Projekt für ihn nicht mehr durchschaubar wird. Er entwickelt ein Mißtrauen gegenüber dem Projekt. Die schlechte Motivation des Kunden führt zu Streitigkeiten und egoistischem Denken auf beiden Seiten. Die Konsenzbildung wird hierdurch sehr erschwert. Durch ein Aufschaukeln kann diese Tatsache bis zum Projektabbruch führen.

Eine sehr häufige Einbeziehung des Kunden bleibt aber auch nicht folgenlos. Durch diese enge Einbeziehung entsteht eine höhere Anzahl von Anforderungsänderungen, da der Kunde jeden Arbeitsschritt einzeln kontrolliert und daher auch an jedem Arbeitsschritt Änderungen wünscht. Diese Anforderungsänderungen sind aber nicht problematisch, da sie sehr früh auftreten und noch im Rahmen der vorgesehenen Tätigkeit eingefügt werden können. Aus einem häufigen Kontakt zum Kunden entstehen oft enge Beziehungen zum Kunden. Diese sehr engen Kontakte entstehen meist informell und auf Entwicklerebene, also nicht auf der Ebene, auf der sich die offiziellen Kontakte befinden. Diese informellen Kontakte sind meistens hochproduktiv und helfen, das Verständnis für die jeweils "andere" Seite zu erhöhen. Durch dieses gegenseitige Verständnis entsteht eine höhere Motivation beim Kunden und bei den Entwicklern. Eine weitere Folge dieser informellen Kontakte ist eine geringere Anzahl von Anforderungsänderungen, die in den meisten Fällen dann auch noch früher zu Tage kommen [Weltz92]. Man geht von beiden Seiten mehr aufeinander ein und das Verständnis der Entwickler für das Anwendungsumfeld wird besser. Aber diese informellen Kontakte haben auch negative Seiten. Sie verleiten zu undokumentierten und nicht abgesprochenen Änderungen.

Hypothese

Durch nicht dokumentierte und nicht abgesprochene Änderungen kommen Fehler in das System, die später nicht mehr oder nur noch sehr schwer zu korrigieren sind. Dieser Effekt verstärkt sich extrem durch das Ausscheiden von Mitarbeitern, die einen solchen engen Kontakt gepflegt haben. Ein Mitarbeiter, der eine enge Beziehung zum Kunden pflegt, hat also damit auch eine Schlüsselposition inne.

2.2.5 Beziehung zum Senior Management

Die Beziehung zum Senior Management kann in Projekten ausschlaggebend für den Erfolg oder Mißerfolg sein. In [Mandl99] wird beschrieben, daß eine schlechte und zu sel-

tene Beziehung zum Senior Management oft Terminschwierigkeiten nach sich zieht, da dem Senior Management eine Kontrolle des Projektfortschrittes fehlt. Wäre diese Kontrolle vorhanden, ließe sich die eine oder andere Krise oder Terminschwierigkeit eventuell mit Hilfe des Senior Managements verhindern. Eine weitere Auswirkung der Beziehung zum Senior Management ist, daß der Kommunikationsaufwand und damit die Arbeitsbelastung des Projektleiters durch Meetings, Statusberichte usw. in Krisensituationen stark ansteigt. So muß er mehr Zeit für Berichte und Meetings mit dem Senior Management aufwenden, obwohl er gerade in dieser Zeit innerhalb seines Projektteams extrem viel gebraucht würde, um den Fortschritt zu kontrollieren und Krisen aus dem Weg zu räumen. Dies führt oft zu einer starken Überbelastung des Projektleiters, einer der wichtigsten Personen in einer Projektkrise.

Die Beziehung des Projektleiters zum Senior Management ist aber auch von Bedeutung für die Motivation der Mitarbeiter. Je nachdem wie gut sich die Mitarbeiter gegenüber dem Management vertreten fühlen, desto besser wird ihre Beziehung zum Projektleiter werden und damit (nach [Herzb59]) auch ihre Motivation. Ein Projektleiter, der sein Fähnchen also immer in den Wind des Managements hängt, trägt damit zur Motivationsenkung seiner Mitarbeiter bei.

2.2.6 Teamzusammenstellung

Art der Mitarbeiter

Verschiedene Mitarbeiter haben verschiedene Bedürfnisse und müssen deshalb verschieden behandelt werden [Hentze95]. Im Bereich EDV gelten hier besondere Aspekte. Gleicht man die Bedürfnisse eines EDV-Spezialisten [Coug78] mit den Menschenbildern von Schein ab [Hentze95], so erkennt man den EDV-Spezialisten hauptsächlich als selbstverwirklichenden Menschen. Um den maximalen Nutzen zu erzielen, sollte die Tätigkeit also vor allem interessant, ausfüllend und selbstverwirklichend sein. Der typische EDV-Spezialist legt nur einen geringen Wert auf die soziale Komponente einer Tätigkeit, möchte aber für seine Tätigkeit Feedback durch den Führenden erhalten, um sich persönlich zu verbessern. Eine Möglichkeit für eine Führungskraft ist, wie bereits oben erwähnt, sich öfter in einem Mitarbeitergespräch mit seinen Angestellten zu unterhalten und deren Stimmung abzufragen. Diese Gelegenheit sollte man als Führungskraft oder Senior-Entwickler nutzen, um die Ergebnisse mit seinen Mitarbeitern kritisch zu diskutieren, aber nicht um auf seinem Standpunkt zu beharren sondern um den Mitarbeitern ein Mitbestimmungsrecht einzuräumen.

Diese Feststellung deckt sich mit der gesamten Literatur [Hentze95] [Weltz92] und stellt eine Gewichtung der Motivatoren und Hygienefaktoren von Herzberg dar. Für Softwareentwickler sind also die Faktoren Arbeitsinhalt und Anerkennung überdurchschnittlich wichtig.

Hypothese

Aufgaben, die nicht zu diesen persönlichen Zielen beitragen, senken bei Softwareentwicklern die Motivation. Dies gilt im besonderen auch für Langeweile und Nichtstun, also Unterforderung. (siehe auch [Herzb59])

Teamwork

Alle Autoren stimmen darin überein, daß ein gutes Teamwork die Produktivität extrem steigern kann. Ein gutes Team bildet sich nach McDonald [McDona99] aus Spezialisten, die sich gegenseitig ergänzen. Hier kann jeder, klare Aufgabenteilung vorausgesetzt, sein Fachgebiet vertreten und das Team ergänzt sich gegenseitig. Da alle weiteren Faktoren zur guten Teambildung, wie gegenseitiges Verständnis und gute Zusammenarbeit in SESAM voraussichtlich nicht verwirklicht werden können, wird dieses Thema hier nicht weiter vertieft.

Schlüsselpositionen

Die Leistung von Inhabern von Schlüsselpositionen hat eine entscheidende Wirkung auf die Gesamtleistung eines Teams. Oft hängt das Ergebnis eines ganzen Projektes von der Leistung einzelner Personen ab [Weltz92]. Schlüsselpositionen gibt es, wie bereits in der Aspektdefinition beschrieben, jedoch in positiver und negativer Ausprägung. Die Entstehung von Schlüsselpositionen ist jedoch in der Literatur noch nicht beschrieben.

Hypothese

Schlüsselpositionen entstehen durch die besondere Ausprägung von Merkmalen bei bestimmten Personen. Diese Merkmale sind in der restlichen Gruppe oft nur wenig bis gar nicht vorhanden. Typische Merkmale sind:

- Große Erfahrung im Anwendungsbereich
- Lange Projekterfahrung
- Soziale Fähigkeiten
- Überdurchschnittlich gute Beziehung zum Kunden

Im negativen Sinn:

- Dauernde Unzufriedenheit (geringe Motivation)
- Schlechtes Teamwork, verschiedene Arbeitsweisen von Mitarbeitern innerhalb eines Teams

Die Auswirkungen des Vorhandenseins oder Wegfalles von Schlüsselpositionen sind:

positiv: motivationssteigernd und damit produktivitätssteigernd

negativ: motivationssenkend und damit produktivitätssenkend

Diese Hypothese deckt sich wiederum mit den Erkenntnissen von Herzberg.

Hypothese

Das Wegfallen oder Hinzukommen von Schlüsselpositionen durch Kündigung oder Neueinstellung verstärkt die in Kapitel 2.2.3 genannten Faktoren vor allem hinsichtlich des Gesamtprojektes.

2.2.7 Mitarbeiterführung

Auch die Personalführung besitzt viele Zusammenhänge zu anderen Verhaltensaspekten in Softwareprojekten. Thayer [Thaye97] sieht den Hauptzusammenhang darin, daß durch eine richtige Personalführung die Produktivität erhöht werden kann. Die steht im Zusammenhang mit einer erhöhten Motivation. Die einzelnen Aufgaben eines Führenden zur Produktivitätserhöhung bezeichnet Thayer wie folgt:

- Ausräumen von Schwierigkeiten
- Klare Zielvorgaben

- Einfachere Kommunikation
- Gute Teambildung
- Koordination der Aktivitäten
- Motivation des Personals
- Delegieren von Aufgaben
- Lösen von Konflikten

Auch diese Definition stimmt mit der Motivationsdefinition von Herzberg überein, der die Führungstechnik als einen der Hygienefaktoren bezeichnet. Durch die von Thayer benannten Faktoren steigt also die Motivation der Mitarbeiter, was eine höhere Produktivität nach sich zieht.

Thayer stellt des weiteren fest, daß durch eine höhere Produktivität wiederum eine höhere Motivation erreicht wird. Auch dieser Punkt wird von Herzberg bestätigt (Leistungserlebnis als Motivator).

Führungsstil

Eine umfassende Aussage, welcher Führungsstil welche Ergebnisse liefert, kann nicht gegeben werden. In vielen empirischen Untersuchungen wurde bewiesen, daß der jeweils beste Führungsstil von der Situation und den Mitarbeitern abhängig ist. Auswirkungen hat der Führungsstil aber auf die Motivation der Mitarbeiter und damit auch indirekt auf deren Produktivität. Wie bereits in der Aspektdefinition erwähnt, werden für die heutige Zeit eher demokratische Führungsstile empfohlen. Vor allem im Bereich der Softwareentwicklung, in der vor allem hochwertig ausgebildete und eigenständig arbeitende Mitarbeiter eingesetzt werden, sollte diese Empfehlung befolgt werden. Werden die Mitarbeiter zu autoritär geführt, geht deren Selbständigkeit verloren und damit die Motivatoren "Anerkennung" und "Verantwortung", wiederum zwei Faktoren, die laut Herzberg wichtig für die Motivation sind. Jedoch ist gerade bei Softwareentwicklern auch eine gewisse Kontrolle gefragt. Da es sich größtenteils um klar strukturierbare Aufgaben handelt (sofern moderne Projektmanagement- und Entwicklungsmethoden angewendet werden), läßt sich dies auch einfach durchführen. Vorschläge für Hilfsmittel, die zu einem angemessenen Führungsstil in einem Softwareprojekt führen, wären beispielsweise:

- Regelmäßige Meetings, in denen die Entwickler ein gewisses Mitspracherecht besitzen.
- Delegation von wichtigen und verantwortungsvollen Aufgaben.
- Eigenständige Kontrolle des Projektfortschrittes durch die Mitarbeiter, der durch die Führungskraft nur noch in größeren Abständen kontrolliert wird.
- Längerfristige Führung durch Zielvorgaben ("Management by Objectives").
- Indirekte Kontrolle der Mitarbeiter bspw. durch die Frage "Wie geht es voran?" anstelle der Aufforderung, die Ergebnisse zu präsentieren.
- Mitarbeitergespräche, in denen die Führungskraft persönlich auf den Mitarbeiter eingeht.

Einen anderen interessanten Ansatz zeigt Hentze [Hentz95]. Er ist der Ansicht, daß nicht immer der Führungsstil die Leistung beeinflußt, sondern daß es in bestimmten Fällen die Leistung der Mitarbeiter ist, die den Führungsstil des Führenden bestimmt. Ein Team, daß

beispielsweise sehr autonom und selbständig arbeitet, führt eventuell den einen oder anderen Projektleiter dazu, einen sehr demokratischen Führungsstil zu wählen, da sich dies mit der Zeit bewährt.

Über- und Unterforderung

Wie stark eine Über- oder Unterforderung ein Softwareprojekt beeinflusst, hängt sehr stark von der Art der Aufgabe ab. Weltz und Ortmann (1992) zeigen, daß ein Mitarbeiter, der sich mit einer interessanten Aufgabe beschäftigt, viel eher bereit ist Überstunden in Kauf zu nehmen. Ist die Aufgabe nicht befriedigend, so resultiert Überforderung in einem Motivationsverlust und in Streßsymptomen oder einer Kündigung. Unterforderung führt dazu, daß Mitarbeiter lethargisch werden und sich nicht mehr mit der Firma identifizieren können, da sie (scheinbar) nicht wichtig genug sind, um genügend Aufgaben zu bekommen. Auch dies führt zu einem starken Motivationsabfall.

Auch hier läßt sich eine klare Affinität zu Herzberg feststellen. Ist die Aufgabe interessant und ausfüllend (Arbeitsinhalt, Verantwortung, Leistungserlebnis), so steigt die Motivation. Ist sie dies nicht und müssen trotzdem Überstunden aufgewendet werden, so sinkt die Motivation und resultiert in diesem Fall in Unzufriedenheitsauswirkungen auf die Gesundheit (Streßsymptome) und als letzten Ausweg in der Kündigung.

Kapitel 3:

Umsetzung der Aspekte

Nachdem im vorherigen Kapitel die Aspekte betrachtet wurden, die in Softwareprojekten allgemein auftreten, mußte in einem nächsten Schritt eine Auswahl getroffen werden, welche dieser Aspekte in das QS-Modell integriert werden können. Hierfür werden zuerst Kriterien festgelegt, die eine Auswahl dieser Aspekte ermöglichen. Die definierten Aspekte wurden dann systematisch betrachtet und anhand der Kriterien auf ihre Tauglichkeit für das QS-Modell und die zugrunde liegenden Konzepte geprüft. Im nächsten Schritt wurden die ausgewählten Aspekte modelliert, so daß sie im QS-Modell umgesetzt werden konnten.

3.1 Erarbeitung geeigneter Kriterien zur Auswahl der Aspekte

Um aus der Vielzahl der in Kapitel 2 definierten Aspekte eine Auswahl treffen zu können, müssen Kriterien festgelegt werden, anhand derer eine Bewertung der Aspekte möglich ist. Bei der Definition der Kriterien wurde darauf geachtet, daß die Kriterien möglichst alle Aspekte, die zu betrachten sind, umfassen, aber gleichzeitig möglichst disjunkt sind. Da dies jedoch in der Realität nicht immer erreicht werden kann, sind auch hier gewisse Überschneidungen möglich. Die Kriterien lassen sich auch als Ziele definieren, die mit den ausgewählten Aspekten erreicht werden sollen:

1. Ziel: Lerneffekt

Das Hauptziel des gesamten QS-Modelles ist die Ausbildung von Projektleitern. Daher muß jeder Aspekt, der in das QS-Modell integriert wird, einen Lerneffekt haben. Es muß

also klar definiert sein, was dem Spieler beigebracht werden soll und wodurch dies erfolgt. Die einzelnen Ausbildungsziele, die hinter dem QS-Modell stehen, sind in [Drapp00] beschrieben:

- Lernen der eigenständigen Planung und Führung eines Softwareprojektes
- Konfrontation mit typischen Schwierigkeiten
- Einnahme der typischen Rolle des Projektleiters
- Lernen der Zusammenhänge zwischen Projektsteuerung und Projektergebnissen
- Lernen von quantitativen Zusammenhängen in Softwareprojekten

An diesen Zielen hat sich auch in dieser Arbeit die Auswahl der zusätzlichen Aspekte orientiert.

2. Ziel: Realitätsnähe

Dieses Ziel wird bereits in [Drapp00] als Unterpunkt der Ausbildungsziele genannt. Da die dortige Definition den Punkt aber noch nicht vollständig beschreibt, wird er hier erweitert und als eigenes Kriterium aufgeführt. Zusätzlich zu den von Drappa [Drapp00] beschriebenen Aspekten wird der Spieler durch eine stärkere Realitätsnähe mehr in das Spiel hineinversetzt. Eine hohe Realitätsnähe hilft also dem Spieler dabei, im Spiel erlernte Effekte im nachhinein in realen Projekten wiederzuerkennen und anzuwenden. Ein weiterer zusätzlicher Punkt ist, daß die Bedienbarkeit für den Spieler durch eine hohe Realitätsnähe erheblich einfacher wird, da die meisten Situationen aus der Praxis bekannt sind. Es müssen also keine künstlichen Situationen einstudiert und neu erlernt werden. Jeder Spieler kann den Simulator direkt anwenden.

3. Ziel: Granularität

Die Granularität muß über das gesamte Modell einheitlich sein. Eine stark schwankende Granularität erschwert es einem Spieler extrem, seine Möglichkeiten in gewissen Situationen zu begreifen. Wird in bestimmten Bereichen des Modells auf einer sehr hohen Granularitätsstufe modelliert, in anderen Bereichen dagegen sehr fein, so verliert der Spieler den Überblick über die Handlungsmöglichkeiten in gewissen Situationen. Ein Ausbildungsziel aus [Drapp00] ist, daß der Spieler alle Handlungen selbständig vornehmen soll. Dies kann er aber nur, wenn ihm bekannt ist, welche Möglichkeiten er dazu überhaupt besitzt. Des weiteren dürfen im Modell keine Effekte vorkommen, die so fein modelliert sind, daß der Spieler sie nicht erkennt.

4. Ziel: Umsetzbarkeit

Da das Modell in der Praxis in einem Projektmanagement-Simulator eingesetzt und für die Ausbildung von Projektleitern genutzt werden soll, müssen die modellierten Effekte natürlich auch mit einem realistischen Aufwand umsetzbar sein. Natürlich soll dies nicht bedeuten, daß hier eventuelle Mühe gescheut wird, gewisse Aspekte zu implementieren. Es muss jedoch kritisch betrachtet werden, ob das QS-Modell eine Implementierung eines Effekte überhaupt zuläßt und wenn ja, ob die Mittel und Aufwände, die zur Implementierung nötig sind, gerechtfertigt sind. Manager großer Firmen würden diesen Punkt heutzutage vielleicht als "Return on Effort (RoE)" bezeichnen.

Die in Kapitel 2 definierten Aspekte werden nun anhand der erarbeiteten Kriterien auf die Relevanz für das QS-Modell überprüft. Hierbei muß, wie es bei einer Auswahl anhand von verschiedenen Kriterien nötig ist, sicherlich der eine oder andere Kompromiß geschlossen werden. Der Hauptfokus bei der Betrachtung liegt in jedem Fall auf den Zielen Lerneffekt und Granularität. Die Kriterien Realitätsnähe und Umsetzbarkeit dürfen jedoch nicht außer Acht gelassen werden, sondern greifen als unterstützende Kriterien in den Auswahlprozeß ein.

Um die in diesem Kapitel erarbeiteten Ergebnisse im weiteren Verlauf der Arbeit, vor allem während der Quantifizierung, gezielt und einfach weiterverwenden zu können, wird die Modellierung hier bereits teilstrukturiert. Die Modellierung erfolgt für alle Aspekte nach demselben Schema: Zuerst werden die Hauptzusammenhänge eines Aspektes in klare Thesen zusammengefaßt. Für all diese Thesen lassen sich während der Quantifizierung sehr einfach Berechnungsvorschriften erstellen.

Alle Zusammenhänge, die in diesem Kapitel beschrieben werden, existieren natürlich auf beiden Seiten. Einmal in dem Aspekt, der diesen Zusammenhang auslöst (als Output) und einmal in dem Aspekt auf den sich der Zusammenhang auswirkt (als Input). Um die Übersichtlichkeit zu wahren, werden die Zusammenhänge nicht bei beiden Aspekten aufgeführt, sondern dem Aspekt zugeordnet, in dem sie die größere Bedeutung besitzen.

Ein weiterer wichtiger Punkt im Rahmen der Modellierung ist die Frage, inwieweit der Spieler einen Aspekt erlebt oder beobachten kann und inwieweit er darauf Einfluß nehmen kann. Hierfür wird für jeden Aspekt in einem getrennten Abschnitt die Schnittstelle des Aspektes zum Spieler definiert, in dem diese Fragen beantwortet werden.

3.2 Motivation der Entwickler

Für diese Arbeit muß die Motivation als zentraler Aspekt auf jeden Fall umgesetzt werden. Da sich nach Herzberg die Motivation aus verschiedenen Aspekten zusammensetzt, soll diese Struktur auch in dieser Arbeit beibehalten werden, um die Nachvollziehbarkeit zu gewährleisten.

Einige der Motivationsaspekte werden nicht realisiert, da sie anhand der oben erwähnten Kriterien nur eine untergeordnete Rolle spielen. Nicht berücksichtigt werden folgende Aspekte aus der Herzbergschen Definition (siehe Kap. 2.2.1):

- Beruflicher Aufstieg
- Unternehmenspolitik
- Entwicklungsmöglichkeiten
- Beziehung zu Unterebenen
- Status
- Arbeitsplatzsicherheit
- Entlohnung
- Verantwortung
- Führungstechniken
- Schlüsselpositionen

Alle anderen von Herzberg definierten Motivationsfaktoren werden im Rahmen dieser Arbeit in SESAM zumindest indirekt realisiert.

Im Rahmen der Modellierung der einzelnen Motivationsaspekte zeigte sich, daß eine Modellierung wie sie Herzberg beschrieben hat, aus mehreren Gründen nicht sinnvoll ist:

- Für einige der sehr detailliert beschriebenen Einzelaspekte wurden keine ausreichend fundierten Daten gefunden.
- Eine Modellierung aller Aspekte ergab ein Vielzahl von Zusammenhängen, die zu einer sehr komplizierten Quantifizierung und als Folge dessen zu einer kaum möglichen Validierung geführt hätten.
- Durch die schwierige Trennung zwischen den einzelnen Aspekten ergaben sich viele zyklische Zusammenhänge, die eine Quantifizierung unmöglich machten.

Zur Verdeutlichung dieser Probleme ist hier ein Bild der daraus resultierende Modellierung aufgezeichnet:

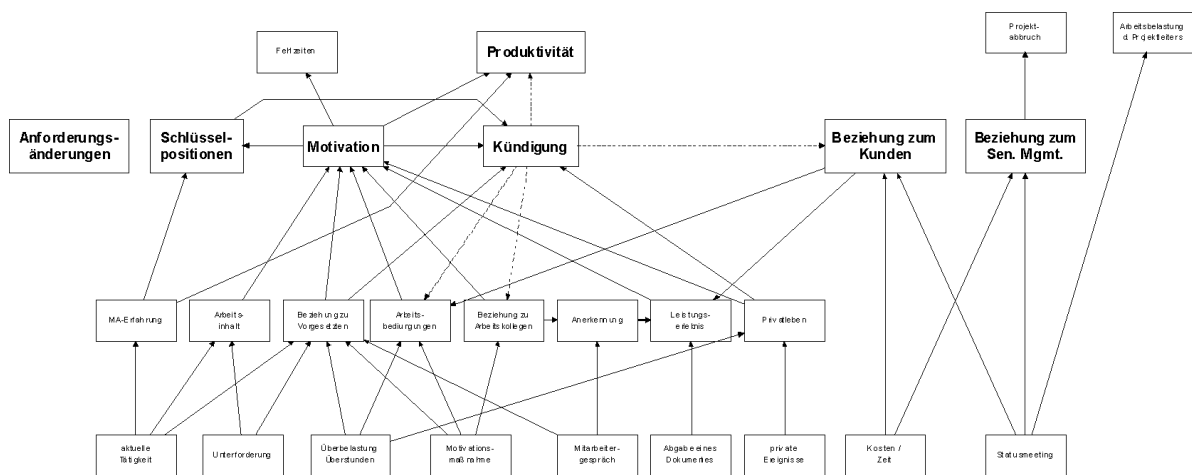


Abbildung 2: Modellierung der Zusammenhänge nach Herzberg(1958)

Aus diesem Grund wurde die Entscheidung getroffen, die einzelnen Motivationsaspekte nur indirekt in die Modellierung einfließen zu lassen. Die einzelnen Ereignisse und Maßnahmen, die die Gesamtmotivation eigentlich über die Motivationsaspekte beeinflussen, wirken sich nun direkt auf die Gesamtmotivation aus. Die einzelnen Motivationsaspekte werden jedoch nicht unterschlagen. Sie wirken sich im späteren Verlauf der Arbeit durch die Quantifizierung indirekt auf die Gesamtmotivation aus und werden für die Begründung der nun verkürzten Zusammenhänge auch im Rahmen der Modellierung verwendet. Die vereinfachte Modellierung stellt sich nun wie folgt dar:

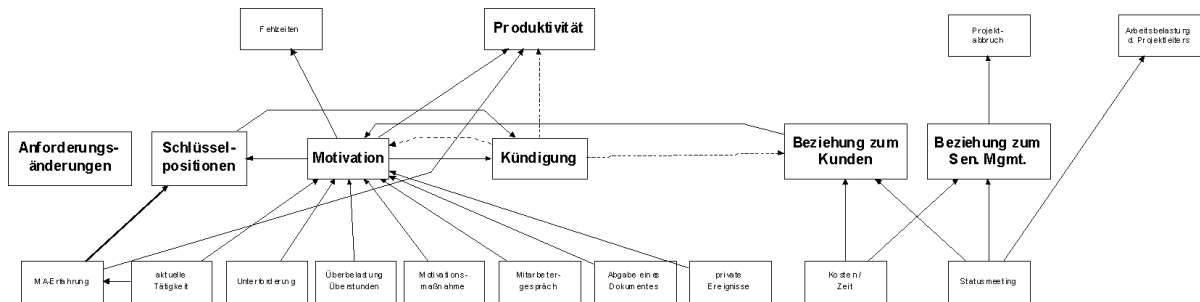


Abbildung 3: Verkürzte Modellierung der Zusammenhänge

Im weiteren Verlauf dieses Kapitels wird diese Modellierung der Motivation nun ausführlich beschrieben.

3.2.1 Gesamtmotivation der Entwickler

Der Aspekt *“Gesamtmotivation der Entwickler”* hat innerhalb des QS-Modelles eine zentrale Bedeutung für viele andere Aspekte. Anhand der Gesamtmotivation werden die Auswirkungen der Motivation eines Mitarbeiters realisiert. Es existiert eine Standardmotivation, also ein Wert, der eine durchschnittliche Motivation repräsentiert. Liegt der aktuelle Wert der *Gesamtmotivation* darüber, so ist der Mitarbeiter motivierter als der Durchschnitt, liegt er darunter, so ist er unmotivierter.

Die Berechnung der Gesamtmotivation setzt sich aus vielen einzelnen Punkten zusammen, die jedoch nicht alle gleich gewichtet werden. So ist beispielsweise bekannt, daß gerade EDV-Spezialisten nur einen geringen Wert auf die sozialen Komponenten der Motivation, wie Beziehung zu Arbeitskollegen, Beziehung zu Vorgesetzten oder das Privatleben legen. Erhöhten Wert legen sie dagegen auf Aspekte, die die Arbeit selbst betreffen, wie die Arbeitsbedingungen und den Arbeitsinhalt. Entsprechend mehr oder weniger werden diese Aspekte bei der Berechnung der Gesamtmotivation berücksichtigt.

Zusammenhänge zu weiteren Aspekten

- Eine interessante Tätigkeit wirkt sich auf die Motivation aus. Die Frage, ob eine Arbeit interessant ist, wird auf folgenden Zusammenhang reduziert: Hat ein Mitarbeiter auf einem bestimmten Gebiet mittlere oder hohe Erfahrung, wird davon ausgegangen, daß die Tätigkeit für ihn interessant ist. Dies steigert die Motivation des Mitarbeiters. Im Gegensatz dazu senkt eine Tätigkeit, die einen Mitarbeiter nicht interessiert, die Motivation.
- Für Nichtstun oder Überstunden macht der Mitarbeiter seinen Vorgesetzten verantwortlich, was sich auf die Beziehung zum Vorgesetzten und auf den Arbeitsinhalt auswirkt und damit auf die Motivation. Überstunden wirken sich zusätzlich negativ auf das Privatleben des Mitarbeiters und damit auch auf seine Motivation aus.
- Eine Motivationsmaßnahme wirkt sich positiv auf die Motivationsfaktoren Beziehung zu Vorgesetzten, Arbeitsbedingungen und Beziehung zu Arbeitskollegen aus und steigert folglich die Motivation.

- Durch ein Mitarbeitergespräch bekommen Führungskraft und Angestellter ein besseres Verständnis füreinander und der Mitarbeiter bemerkt, daß sich die Führungskraft für seine Belange interessiert. Damit steigt die Motivation des Mitarbeiters.
- Zufällige externe Ereignisse wirken sich auf den Motivationsfaktor Privatleben und dadurch auch auf die Motivation aus.
- Besitzt ein Mitarbeiter eine Motivation, die einen gewissen Prozentsatz unter bzw. über der Standardmotivation liegt, so sind seine Produktivität, seine Korrekturleistung und seine Prüfleistung auch um einen davon abhängigen Prozentsatz niedriger bzw. höher.
- Eine niedrige Motivation führt zu vermehrten Krankentagen und Fehlzeiten.
- Als letztes Mittel im Rahmen einer sehr niedrigen Motivation sieht der Mitarbeiter nur die Kündigung. Fällt die Motivation also unter einen bestimmten Schwellenwert, so reicht er seine Kündigung ein.
- Eine Kündigung wirkt sich negativ auf die Beziehung zu den Arbeitskollegen des Mitarbeiters und damit auch auf deren Motivation aus.

Der zeitliche Ablauf einer ständig schlechter werdenden Gesamtmotivation ist wie folgt [Herzb59]:

1. Produktivitätsabfall
2. Eventuell ein vom Mitarbeiter initiiertes Mitarbeitergespräch
3. Fehlzeiten
4. Bekanntgabe der Kündigung -> Kündigung

Die einzelnen Stufen werden jeweils durch bestimmte Schwellenwerte der Gesamtmotivation ausgelöst. Von den Stufen 1-3 läßt sich durch eine Motivationssteigerung des Mitarbeiters wieder herabkommen. Ist die Kündigung bekanntgegeben, so läßt sich der Mitarbeiter nicht mehr davon abbringen und wird unwiderruflich kündigen. Da die Mitarbeiter verschieden sind, finden die Punkte 2 und 4 nicht im Rahmen jedes Mitarbeiters statt. Ein Mitarbeiter fällt bei einer schlechten Gesamtmotivation durch Fehlzeiten auf, ein anderer verlangt ein Mitarbeitergespräch. Es kann jedoch auch beides oder nichts davon auftreten.

Schnittstelle zum Spieler

Der Projektleiter in SESAM hat die Möglichkeit, anhand eines Mitarbeitergespräches, der Krankentage, der Produktivität und von Kündigungen die Motivation seiner Mitarbeiter festzustellen:

- *Fehlzeiten:* Aus der Anzahl und der Häufigkeit der Fehltage lassen sich Vermutungen über die Motivation anstellen.
- *Kündigung:* Die Anzahl von Kündigungen und die Tatsache, ob Kündigungen gehäuft oder nicht auftreten, lassen auf die Motivation der Mitarbeiter schließen.
- *Mitarbeitergespräch:* In einem Mitarbeitergespräch hat der Projektleiter die Möglichkeit, mit seinem Angestellten über die positiven und negativen Aspekte der Arbeit zu sprechen. Das Mitarbeitergespräch wird gesondert behandelt.
- *Produktivität:* Der Projektleiter hat theoretisch durch Zeitvergleiche für eine vergleichbare Tätigkeit die Möglichkeit, die Produktivität eines Mitarbeiters zu ermitteln und könnte daraus Rückschlüsse auf die Motivation ziehen. In der Praxis ist dies jedoch wahrscheinlich nicht durchführbar. Auf die Produktivität wirken auch andere

Faktoren wie beispielsweise die Erfahrung ein. Außerdem ist es sicherlich nicht möglich von Hand einen Produktivitätsunterschied von wenigen Prozentpunkten festzustellen.

Durch das regelmäßige Durchführen von Mitarbeitergesprächen und die Aufrechterhaltung einer guten Beziehung der Mitarbeiter zu den Arbeitskollegen, kann der Spieler für einen hohen Wert des Aspektes *Motivation* sorgen.

In einem Mitarbeitergespräch hat der Spieler die Möglichkeit, diese Motivation des Mitarbeiters zu erfahren.

Da sich auch die Art der Tätigkeit und die Über- oder Unterlast eines Mitarbeiters auf die Motivation auswirken, sollte der Spieler für diesbezüglich ausgeglichene Werte sorgen. Weichen diese Werte in einem Projekt vom Standard ab, so kann der Spieler von einer niedrigen Motivation seiner Mitarbeiter ausgehen.

Ein einfacher, jedoch oft kostspieliger Schritt zur Erhöhung der Motivation sind Motivationsmaßnahmen. Diese steigern die Motivation, können aber nicht immer durchgeführt werden und kosten viel Geld und auch Arbeitszeit.

3.2.2 Mitarbeitergespräche

Schon in der Realität ist es sehr schwierig, Informationen über die Motivation der eigenen Mitarbeiter zu bekommen. In einer Simulation eines Projektes ist es jedoch fast unmöglich. Aus diesem Grund wird in das Modell ein Mitarbeitergespräch eingefügt, in welchem sich der Projektleiter einen Überblick über die Motivation seines Mitarbeiters verschaffen kann.

Der Lerneffekt in diesem Aspekt liegt darin, daß sich ein Projektleiter um die Belange seiner Mitarbeiter kümmern muß. Mitarbeitergespräche sind oft die einzige Möglichkeit um die Motivation eines Mitarbeiters herauszufinden. Außerdem tragen sie zu einer Motivationssteigerung der Mitarbeiter bei, da sich der Mitarbeiter von seinem Vorgesetzten beachtet und teilweise auch verstanden fühlt.

Zusammenhänge zu weiteren Aspekten

- Durch ein Mitarbeitergespräch bekommen Führungskraft und Angestellter ein besseres Verständnis füreinander.

Schnittstelle zum Spieler

Führt der Spieler ein Mitarbeitergespräch mit einem Entwickler, so wird ihm am Ende des Gespräches eine Einschätzung der Gesamtmotivation des Mitarbeiters mitgeteilt. Dies soll die Tatsache repräsentieren, daß der Projektleiter während eines realen Mitarbeitergespräches die Möglichkeit hätte, aufgrund seiner Erfahrung und des Verhaltens des Mitarbeiters die Gesamtmotivation einzuschätzen. Da diese Verhaltensaspekte in einem Spiel mit SESAM nicht vermittelt werden können, wird dem Spieler die Motivation mitgeteilt. Hierbei kann es sich um 5 Stufen handeln: *sehr unmotiviert*, *unmotiviert*, *ausgeglichen*, *motiviert*, *sehr motiviert*. Der genaue Wert wird dem Spieler nicht mitgeteilt, da dies auch in einem realen Gespräch nicht der Fall wäre.

3.2.3 Motivationsmaßnahmen

Ein in der Realität oft gebrauchtes Mittel zu Motivationssteigerung und zur Teambildung sind außerberufliche Aktivitäten wie beispielsweise ein gemeinsames Abendessen oder

einfach auch schon ein durch einen Mitarbeiter mitgebrachter Kuchen. Auch im QS-Modell soll es die Möglichkeit geben, auf diese Art und Weise seine Mitarbeiter zusätzlich zu motivieren. Diese Maßnahmen sind wichtige Mittel eines Projektleiters, um die Motivation des Gesamtteams anzuheben. Der Einsatz solcher Maßnahmen soll auch im QS-Modell geübt werden können.

Zusammenhänge zu weiteren Aspekten

- Abhängig von der durchgeführten Motivationsmaßnahme verbessert sich das Verhältnis des Mitarbeiters zu seinem Vorgesetzten und damit die Motivation.
- Abhängig von der durchgeführten Motivationsmaßnahme verbessert sich das Verhältnis des Mitarbeiters zu seinem Kollegen und damit die Motivation.
- Durch eine Auflockerung des Arbeitsalltages mit einer teambildenden Maßnahme empfindet der Mitarbeiter eine Verbesserung der Arbeitsbedingungen und damit der Motivation.

Schnittstelle zum Spieler

Die Durchführung einer Motivationsmaßnahme liegt ganz in der Hand des Spielers. Er hat jederzeit die Möglichkeit, diese durchzuführen.

3.3 Produktivität

Die Produktivität ist einer der wichtigsten Faktoren im bisherigen QS-Modell. Im Rahmen dieser Arbeit wird der Aspekt Produktivität um die Zusammenhänge erweitert, die durch die Realisierung der anderen Aspekte entstehen.

Zusammenhänge zu weiteren Aspekten

- Muß ein neuer Mitarbeiter eingearbeitet werden, sinkt die Produktivität der anderen Teammitglieder während der Einarbeitungszeit. (Bereits in SESAM realisiert)
- Die Erfahrung der Mitarbeiter hat große Auswirkungen auf die Produktivität (bereits in SESAM realisiert).
- Besitzt ein Mitarbeiter eine Motivation, die einen gewissen Prozentsatz unter bzw. über der Standardmotivation liegt, so sind seine Produktivität, seine Korrekturleistung und seine Prüfleistung auch um einen davon abhängigen Prozentsatz niedriger bzw. höher.

Schnittstelle zum Spieler

Die Produktivität ist nicht direkt meßbar. Der Projektleiter muß die Einflußfaktoren (Erfahrung, Kündigungen, Gesamtmotivation) beobachten und kann daraus die Produktivität abschätzen. Eventuell kann auch die Zeit, die für eine Aufgabe benötigt wird, aussagekräftig sein.

3.4 Fluktuation

Die Fluktuation wird in dieser Arbeit durch die Aspekte Kündigung und unvorhersehbare Kündigung repräsentiert. Der Aspekt Kontinuität wurde aufgrund einer zu feinen Granularisierung herausgenommen. Es kann nicht vom Spieler verlangt werden darauf zu achten, wie konstant die Entwickler im Projekt eingesetzt werden. Des weiteren wurden keine so schwerwiegenden Folgen auf das Projekt ausgemacht, die eine Realisierung trotzdem

gerechtfertigt hätten. Der Aspekt ‐Hinzufügen von Mitarbeitern‐ ist bereits im QS-Modell definiert. Hier werden die Regeln von Brooks [Brook74] bereits befolgt, so da dieser Aspekt nicht zustzlich modelliert werden mu.

3.4.1 Kndigung

Die Kndigung eines Mitarbeiters wird bentigt, um den letzten Schritt einer stark gesenkten Motivation in das Modell zu integrieren. Des weiteren haben Kndigungen und das Ausscheiden aus einem Projekt viele Auswirkungen auf andere Aspekte in einem Projekt, die allgemein nicht unbedingt bekannt sind und deshalb in einer Projektsimulation vorhanden sein sollten, um sie dort zu lernen.

Zusammenhnge zu weiteren Aspekten

- Als letztes Mittel im Rahmen einer sehr niedrigen Motivation sieht der Mitarbeiter nur die Kndigung. Fllt die Motivation also unter einen bestimmten Schwellenwert, so reicht er seine Kndigung ein.
- Eine Kndigung bedeutet durch die ausfallende Arbeitskraft, die nicht direkt hundertprozentig ersetzt werden kann, eine Mehrbelastung der anderen Mitarbeiter.
- Durch das Ausscheiden eines Mitarbeiters wird das Teamwork verndert. Dadurch sinkt die Motivation. Eine lange Projektzugehrigkeit verstrkt diesen Aspekt noch.
- Da nach einer Kndigung ein neuer Mitarbeiter eingearbeitet werden mu, sinkt die Produktivitt der anderen Teammitglieder whrend der Einarbeitungszeit (bereits in SESAM realisiert).

Schnittstelle zum Spieler

Da eine Kndigung durch schlechte Motivation entsteht, hat der Spieler die oben beschriebenen Mglichkeiten, eine Kndigung anhand von schlechter Motivation zu erkennen. Durch die Erhhung der Motivation seiner Mitarbeiter bekommt er die Mglichkeit einer Kndigung vorzubeugen, sofern er den Motivationsschwund rechtzeitig bemerkt hat. Hierdurch wird dem Spieler klar vermittelt, welche Auswirkungen eine niedrige Motivation auf die Kndigungen seiner Mitarbeiter haben kann.

3.4.2 Unvorhersehbare Kndigung

In der Realitt geschehen immer Dinge, die niemand vorraussehen kann. Oft sind diese Dinge durch den Projektleiter berhaupt nicht vorhersehbar, da sie im privaten Bereich des Mitarbeiters, und damit auerhalb der Modellierung des QS-Modells, geschehen. Um dem QS-Modell mehr Realittsnhe zu geben und den Spieler auf solche Ereignisse vorzubereiten, werden unvorhersehbaren Kndigungen in das Modell integriert. Der Grund, warum unvorhersehbare Kndigungen auftreten wird nicht modelliert. Es wird davon ausgegangen, da solche Kndigungen mit einer bestimmten Wahrscheinlichkeit zufllig auftreten.

Zusammenhnge zu weiteren Aspekten

- Zufllige externe Ereignisse knnen sich auf die Motivation auswirken.
- Externe Ereignisse knnen eine solch groe Auswirkung auf die Motivation haben, da der Mitarbeiter sofort eine Kndigung einreicht.

Schnittstelle zum Spieler

Unvorhersehbare Kündigungen sind, wie der Name bereits suggeriert, nicht im voraus zu erkennen. Der Spieler hat keine Möglichkeit etwas dagegen zu tun. Findet eine solche Kündigung statt, so wird der Spieler darüber informiert und der Entwickler verläßt das Projekt. Um eine Abgrenzung zur Kündigung durch schlechte Motivation zu erreichen, wird dem Spieler mitgeteilt, daß es sich um eine unvorhersehbare Kündigung handelt. Im Gegensatz zum vorherigen Abschnitt wird der Spieler hier mit der Tatsache konfrontiert, daß auch Effekte auf die er keinen Einfluß hat, dazu führen können, daß Mitarbeiter völlig unabhängig von ihrer Motivation plötzlich kündigen.

3.5 Beziehung zum Kunden

Auch die Beziehung zum Kunden wird gegenüber der Relativität etwas eingeschränkt. So fällt der sicherlich interessante Punkt "zu gute Beziehung zum Kunden" weg, da hier die Granularität zu fein würde. Ein solcher Punkt ist außerdem vom Spieler vorraussichtlich nicht einfach zu erfassen und würde vermutlich mehr verwirren als helfen. Des weiteren wird die Art der Beziehung zum Kunden nicht modelliert. Die Qualität einer Beziehung läßt sich nur sehr schwer messen, da sie in Gesprächen aufgrund von Gesten und Formulierungen häufig persönlich interpretiert wird. Würden in das Modell einzelne Gesten in Gesprächen integriert, so wäre darüberhinaus die Forderung nach einer konstanten Granularität auf das Höchste verletzt. Eine Berücksichtigung dieses Aspektes ist daher nicht sinnvoll.

Integriert wird dagegen der Aspekt, daß der Kunde in das Projekt einbezogen werden sollte. Hierfür werden Statusmeetings eingeführt, die wiederum einen Projektstatus aus Sicht des Kunden und eine Motivation des Kunden benötigen.

3.5.1 Projektstatus (Kunde)

Der Projektstatus wird für die Berechnung der Motivation des Kunden benötigt und errechnet sich, analog zum Projektstatus aus der Sicht des Senior Managements (s.u.), aus den Abweichungen zu den Aufwänden, die in Boehm definiert wurden. Zusätzlich hat der Kunde aber durch Reviews, an denen er teilnimmt, einen Einblick in die Qualität der bisherigen Ergebnisse. Dieser Faktor muß also in die Berechnung des Projektstatus mit einfließen.

Zusammenhänge zu anderen Aspekten

- Die Motivation des Kunden hängt vom Projektstatus(Kunde) ab.

Schnittstelle zum Spieler

Der Projektstatus berechnet sich analog zum Projektstatus des Senior Managements, aus dem Fortschritt des Projektes. Der Spieler hat also auch hier nur einen indirekten Einfluss darauf, indem er das Projekt voranbringt. Einen schlechten Projektstatus bekommt der Spieler in einem Statusmeeting mitgeteilt.

3.5.2 Motivation des Kunden

Die Beziehung zum Kunden hat in den meisten Projekten einen sehr starken Einfluß auf den Erfolg oder Mißerfolg. Sie wird hier als Motivation des Kunden bezüglich des Projektes modelliert. Des weiteren harmonisiert dieser Aspekt sehr gut mit den anderen Aspekten,

die im Modell realisiert werden und zeigt somit sehr gut die Fehler oder richtigen Entscheidungen, die an anderen Stellen im Projekt getroffen wurden.

Zusammenhänge zu weiteren Aspekten

- Die Motivation des Kunden hängt vom Projektstatus aus Kundensicht ab.
- Je weiter die Motivation des Kunden sinkt, desto häufiger werden Statusmeetings einberufen.
- Sinkt die Motivation des Kunden unter einen bestimmten Schwellwert, so führt dies zum Projektabbruch.

Schnittstelle zum Spieler

Führt der Spieler ein Statusmeeting mit dem Kunden durch, so wird ihm abschließend ein ungefährender Wert über die Motivation des Kunden ausgegeben. Hier wird davon ausgegangen, daß - wie im Mitarbeitergespräch - der Projektleiter aufgrund seines persönlichen Kontaktes, der innerhalb von SESAM nicht realisierbar ist, eine Einschätzung über die Kundenmotivation bekommen hätte.

3.5.3 Statusmeetings mit dem Kunden

Statusmeetings sind eine Aufgabe, mit der sich jeder Projektleiter zu befassen hat. Oft verbrauchen sie einen beträchtlichen Anteil seiner Arbeitszeit. In Statusmeetings möchte der Kunde über den aktuellen Stand des Projektes informiert werden. Gerät das Projekt in eine Krise, so wird der Kunde verlangen, Statusmeetings häufiger abzuhalten. Dies kann für den Projektleiter dazu führen, daß er sich fast nur noch mit Statusmeetings beschäftigen muß. Gerade in der schwierigen Situation einer Projektkrise benötigt er als wichtigste Person im Projekt mehr Zeit zur besseren Koordination, wird aber gleichzeitig durch viele Statusmeetings stark beansprucht. Dies entspricht der Realität und soll deshalb dem Spieler deutlich gemacht werden.

Zusammenhänge zu weiteren Aspekten

- Die Zeit zwischen zwei Statusmeetings hängt von der Motivation des Kunden ab

Schnittstelle zum Spieler

Statusmeetings werden durch den Kunden einberufen. Hierfür benötigt der Projektleiter einen Tag für die Vorbereitung und Durchführung des Meetings. In dieser Zeit hat der Spieler keine Möglichkeit, Befehle einzugeben und das Projekt zu steuern. Am Ende eines Statusmeetings bekommt der Spieler die Motivation des Kunden mitgeteilt und den Termin für das nächste Meeting, damit er sich darauf einstellen kann.

3.6 Beziehung zum Senior Management

Die Beziehung zum Senior Management wird in diesem Modell auf das Durchführen von Statusmeetings reduziert. Da davon ausgegangen wird, daß alle Beteiligten vernünftig miteinander umgehen, werden diese Statusmeetings als positiv für den Projektverlauf angesehen, weil beide Parteien auf jeden Fall mehr Verständnis füreinander erlangen und der Projektleiter durch Ratschläge und Tips unterstützt und angeregt wird..

Die Beziehung zum Senior Management bzw. die Zeit, die durch Statusmeetings verloren geht, ist bisher im QS-Modell nicht realisiert. Wie bereits oben beschrieben, ist es in der Praxis so, daß vor allem in Krisenzeiten der Projektleiter oft mehr damit beschäftigt ist,

irgendwelchen Managern den aktuellen Status zu berichten, als sein Projekt zu koordinieren. Diese Tatsache wird auch hier in das Modell eingefügt und somit dem Spieler beigebracht. Die Beziehung zum Senior Management wird analog zur Beziehung zum Kunden realisiert. Auch hier werden die Aspekte Projektstatus (Senior Management), Motivation des Senior Managements und Statusmeeting (Senior Management) modelliert. Da sich alle drei Aspekte analog zu den Aspekten der Beziehung zum Kunden verhalten, werden diese hier nicht separat beschrieben.

Zusammenhänge zu weiteren Aspekten

- Die Motivation des Senior Managements hängt vom Projektstatus aus Sicht des Senior Managements ab.
- Je weiter die Motivation des Senior Managements sinkt, desto häufiger werden Statusmeetings einberufen.
- Sinkt die Motivation des Senior Managements unter einen bestimmten Schwellwert, so führt dies zum Projektabbruch.

Schnittstelle zum Spieler

Die Beziehung zum Senior Management bzw. die Motivation des Senior Managements bekommt der Spieler im Rahmen eines Statusmeetings mitgeteilt. Außerdem beeinflusst sie die zeitlichen Abstände zwischen den Statusmeetings. Der Projektstatus berechnet sich aus dem Fortschritt des Projektes. Der Spieler hat also nur einen indirekten Einfluß darauf, indem er das Projekt voranbringt. Einen schlechten Projektstatus bekommt der Spieler in einem Statusmeeting mitgeteilt.

3.7 Teamzusammenstellung

Es ist sehr schwierig für den Bereich Teamzusammenstellung geeignete Aspekte zu finden, die sich realisieren lassen. Aspekte die sich sehr stark auf zwischenmenschlicher Ebene bewegen, sind in einem Simulator nur sehr schwer und mit einem hohen Detaillierungsgrad zu implementieren. Aus diesem Grund wurden die verschiedenen Menschenbilder und Schlüsselpositionen nicht implementiert. Um aber die Wichtigkeit eines guten Teams und die Tatsache, daß in den Projekten eigenständige Menschen arbeiten nicht zu vernachlässigen, wurden die Aspekte Motivationsmaßnahmen (s.o.), veränderliche Erfahrungen der Mitarbeiter und Fehlzeiten in das Modell aufgenommen.

3.7.1 Veränderliche Erfahrungen

Die Erfahrung der Mitarbeiter ist in SESAM bereits realisiert. Diese Realisierung sollte jedoch etwas angepaßt werden. Der bisherigen Realisierung [Drapp00] muß hinzugefügt werden, daß sich die Erfahrung der Mitarbeiter während eines Projektes durchaus ändern kann. Die bisherige statische Modellierung der Erfahrung wird so erheblich realistischer. Des weiteren wird eine Projekterfahrung hinzugefügt. So kann die Zugehörigkeit in einem Projekt durchaus als Erfahrung angesehen werden, da der Projektmitarbeiter die Gegebenheiten und Prozesse des Projektes erheblich besser kennt, je länger im Projekt tätig ist.

Zusammenhänge zu weiteren Aspekten

- Durch das Ausüben einer Tätigkeit erhöht sich die Erfahrung in diesem Bereich.
- Die Projekterfahrung berechnet sich aus der Länge der Projektzugehörigkeit.

- Kündigt ein Mitarbeiter mit einer großen Projekterfahrung, so wirkt sich dies auf die Motivation der anderen Mitarbeiter aus

Alle Einzelerfahrungen werden mit jedem Zeitschritt, in dem der Mitarbeiter im Projekt mit einer der Erfahrung entsprechenden Tätigkeit beschäftigt ist, um einen gewissen Wert erhöht:

In [Drapp00] ist angegeben, welche Erfahrungen sich auf die Produktivität welcher Tätigkeiten auswirken. Umgedreht lässt sich also feststellen, welche Tätigkeiten Einfluß auf welche Erfahrungen haben müssen. Welche *Erfahrungstage* durch welche Tätigkeit erhöht werden ist in folgender Tabelle festgehalten:

Tätigkeit	Erfahrung Anwendungs- -gebiet	Erfahrung Spezifizieren	Erfahrung Entwerfen	Erfahrung Codieren	Erfahrung Notation
Analysieren	X				
Spezifizieren	X	X			X
Entwerfen			X		X
Module spezifizieren					X
Codieren				X	X
Integrieren				X	X
Handbuch schreiben					X

Tabelle 3: Auswirkung der aktuellen Tätigkeiten auf die Erfahrungen der Mitarbeiter

Schnittstelle zum Spieler

Der Spieler kann sich jederzeit die Erfahrung eines Mitarbeiters durch das Aufrufen der Informationen des Mitarbeiters anzeigen lassen. Auswirkungen der höheren Erfahrung sind eine höhere Motivation und eine höhere Arbeitsleistung.

3.7.2 Fehlzeiten

Überraschende Fehlzeiten durch Krankheit von Mitarbeitern gehören zu einem Softwareprojekt wie die Programmierung. Des weiteren ist eine bekannte Auswirkung von schlechter Motivation die Erhöhung des Krankenstandes. Um diesem Aspekt Tribut zu zollen und dem Spieler eine Möglichkeit zu geben, diesen Zusammenhang zu erlernen um so eine niedrige Gesamtmotivation auch ohne Mitarbeitergespräche zu erkennen, werden Fehlzeiten, die von der Motivation abhängen in das QS-Modell integriert. Außerdem wird der Spieler hiermit auf das Problem aufmerksam gemacht, daß Mitarbeiter (häufig auch in kritischen Projektphasen) einfach krank werden, ohne daß dies im voraus für ihn erkennbar gewesen wäre. Eine Situation mit der jeder Projektleiter zumindest rechnen sollte.

Zusammenhänge zu weiteren Aspekten

- Eine niedrige Motivation führt zu vermehrten Krankentagen und Fehlzeiten.

Schnittstelle zum Spieler

Erscheint ein Mitarbeiter nicht zur Arbeit, so wird der Spieler am Anfang eines Tages darauf hingewiesen. Des Weiteren wird dem Spieler die Länge der Fehlzeit mitgeteilt. Erscheint ein Mitarbeiter zurück am Arbeitsplatz, so wird auch dies dem Spieler mitgeteilt. Häufige Fehlzeiten können, müssen aber nicht, ein Indikator für eine schlechte Motivation sein. Fehlt ein Mitarbeiter häufig, sollte der Spieler daher auf jeden Fall ein Mitarbeitergespräch einberufen, allein schon um eine niedrige Motivation zu entdecken oder ausschließen zu können. Da hohe Motivationen die Fehlzeitenwahrscheinlichkeit senken, sollte es der Spieler anstreben, seine Mitarbeiter zu einer hohen Motivation zu führen.

3.8 Mitarbeiterführung

Auch die Mitarbeiterführung kann aufgrund des hohen Anteils an zwischenmenschlichen Beziehungen nur in geringem Maße in das Modell integriert werden. Mitarbeiterführung besteht heute überwiegend aus der Vermeidung von Problemen und der Unterstützung der Mitarbeiter. Der zwischenmenschliche Aspekt ist hier sehr wichtig. Aus diesem Grund konnte der Punkt Führungsstil nicht weiter verfolgt werden. Um die Mitarbeiterführung doch noch integrieren zu können, wurde der Aspekt Überstunden in das Modell aufgenommen.

3.8.1 Überstunden

Ein häufiges Mittel zur kurzfristigen Personalerhöhung sind die Überstunden. Überstunden werden in der aktuellen Zeit immer häufiger und sollten deshalb auch in einem Projektmanagement-Simulator vorhanden sein. Im Rahmen der Realisierung wird es möglich sein, für eine befristete Zeit Überstunden anzuordnen. Natürlich nicht ohne den Hintergedanken, auch die negativen Folgen von Überstunden erkennbar zu machen.

Zusammenhänge zu weiteren Aspekten

- Überstunden empfindet der Mitarbeiter ab einer bestimmten Anzahl als negativ in Bezug auf die Arbeitsbedingungen und die Freizeit und damit auf die Motivation.

Schnittstelle zum Spieler

Die Überforderung wird durch das Anordnen von Überstunden realisiert. Der Projektleiter bekommt die Möglichkeit, für einen gewissen Zeitraum Überstunden anzuordnen, die dann von den Mitarbeitern widerspruchslos ausgeführt werden und stets mit dem aktuellen Stundensatz des Entwicklers plus eines Überstundenzuschlags berechnet werden. Ein Wert, der sich aus der Häufigkeit von Überstunden berechnet, wird für jeden Entwickler gespeichert.

Ein ähnlicher Wert existiert für die Unterforderung. Hat ein Mitarbeiter nichts zu tun, so wird das in diesem Wert festgehalten.

Die Arbeitsbelastung wird direkt vom Projektleiter gesteuert. Er kann Überstunden anordnen oder dem Mitarbeiter nichts zu tun geben, so daß er unterfordert ist. Eine Rückmeldung über einen zu hohen Wert erhält der Projektleiter in einem Mitarbeitergespräch oder den Auswirkungen der daraus resultierenden schlechten Gesamtmotivation wie bspw. Fehlzeiten.

3.9 Anforderungsänderungen

Der Aspekt der Anforderungsänderungen ist ein interessanter Aspekt im Zusammenhang mit Softwareprojekten. In der Literatur wird ihr Stellenwert zwar einerseits sehr hoch eingeschätzt, andererseits existieren aber kaum Untersuchungen, die diese Werte quantifiziert haben. Die Datenbasis ist also sehr schwach. Des Weiteren sind Anforderungsänderungen bereits indirekt in das bisherige QS-Modell integriert. Alle Aufwands- und Produktivitätszahlen beruhen auf realen Projekten, beinhalten also indirekt die Aufwände für Anforderungsänderungen. Für eine Realisierung von Anforderungsänderungen hätte also das gesamte QS-Modell neu quantifiziert und damit auch validiert werden müssen. Dieser Aufwand ist im Rahmen dieser Arbeit nicht aufzubringen.

Kapitel 4:

Quantifizierung des Modells

Bisher wurden die für das Modell identifizierten Aspekte nur qualitativ definiert. Um ein vollständiges Modell zu schaffen, mußten diese Aspekte auch quantitativ beschrieben werden.

Wie auch Drappa [Drapp00] berichtet, ist die Parametrisierung eines solchen Modells sehr schwierig, da bisher nur wenige empirische Studien vorliegen, die relevante Daten über Zusammenhänge in Softwareprojekten beinhalten. Bisherige Studien beschäftigen sich vor allem mit gut meßbaren Effekten wie der Produktivität oder der Fehlerrate, nicht aber mit Verhaltens- oder Zufallsaspekten, die um ein vielfaches schwerer zu erfassen und zu messen sind. Die geringe Anzahl von empirisch fundierten Daten müssen in dieser Arbeit also verstärkt durch Hypothesen unterstützt werden.

Trotz dieser Tatsache müssen bei der Quantifizierung die Regeln beachtet werden, die im voraus für das gesamte Modell festgelegt wurden [Drapp00]:

- Die Daten müssen für die für das QS-Modell gewählte Projektklasse gültig sein.
- Die Daten sollten, wenn sie denn empirisch belegt sind, auf einer möglichst breiten statistischen Basis beruhen.
- Die zur Erhebung gewählten Metriken müssen mit den im Modell realisierten Metriken übereinstimmen.

Das Kapitel ist wie folgt aufgebaut: Die identifizierten Aspekte wurden parametrisiert, indem die im vorherigen Kapitel entwickelten Zusammenhänge quantifiziert worden sind.

Hierfür wurden zuerst Berechnungsvorschriften für alle Aspekte definiert. Um klar zeigen zu können, auf welchen Daten die Quantifizierung beruht, wird für jeden Aspekt eine Quelle angegeben, aus der die Quantifizierung stammt, sofern die Quantifizierung nicht auf Hypothesen beruht und eine Quelle gefunden werden konnte. Des Weiteren wird angegeben, inwieweit die Werte der zugrunde liegenden Quelle angepaßt wurden, um auf das QS-Modell zuzutreffen.

4.1 Motivation

Die Motivation wird auf einer Skala zwischen -5 und +5 modelliert und aufgrund ihres Zahlenwertes in verschiedene Klassen für die Repräsentation nach außen eingeteilt:

Motivation M (als Real)	Motivation (als Text)
$M \leq -3$	sehr unmotiviert
$-3 < M \leq -1$	unmotiviert
$-1 < M \leq 1$	ausgeglichen
$1 < M < 3$	motiviert
$M \geq 3$	sehr motiviert

Tabelle 4: Einteilung der Motivation in Klassen

Die Motivation wird nach Herzberg durch positive und negative Erlebnisse beeinflusst (s.Kap. 2.2.1). Die Auswirkungen der Ereignisse auf die Motivation schwächen sich im Laufe der Zeit immer mehr ab. Man kann also sagen, daß diese Ereignisse vergessen werden. Dies bedeutet, daß sich die Motivation im Laufe der Zeit einem gewissen Standardwert annähert:

Berechnungsvorschrift

- Fallen keine besonderen Ereignisse an, so nähert sich die Motivation einem Standardwert (=0) an :

$$\text{Motivation} = \text{Motivation} * (1 - \text{Annäherungswert})$$

Ich gehe davon aus, daß ein Mitarbeiter sowohl ein schlechtes als auch ein gutes Erlebnis bezüglich der Motivation nach etwa zwei Monaten wieder “vergessen” hat, daß es also nach zwei Monaten kaum mehr Auswirkungen auf die Motivation hat. Da dieser Zeitraum ungefähr 42 Arbeitstagen entspricht, nähert sich die Motivation etwa 2.4% pro Tag dem Standardwert an:

$$\text{Motivation} = \text{Motivation} * 0.976$$

Alle Ereignisse, die sich auf die Motivation auswirken, sind in ihrer Auswirkung abhängig von der aktuellen Motivation. So wirkt sich ein motivationssteigerndes Ereignis bei einer Person, die bereits hoch motiviert ist, weniger aus, als bei einer nur durchschnittlich motivierten Person. Aus diesem Grund müssen die Auswirkungen eines Ereignisses an die aktuelle Motivation angepaßt werden. Jedes Ereignis besitzt einen bestimmten Wert, um den sich die Motivation bei einem durchschnittlich motivierten Mitarbeiter verändert. Die-

ser Wert ist der Basisänderungswert. Er wird dann durch eine Funktion angepaßt, die die tatsächliche Änderung aufgrund der aktuellen Motivation berechnet. Diese Funktion läßt sich am besten mit einer zusammengesetzten Cosinusfunktion beschreiben:

$$\text{Motivationsanpassungsfaktor} = \begin{cases} 0.5 * (\cos(0.6*x) + 1) & \text{für } x \geq 0 \\ 0.25 * (\cos(0.6*x)) + 3 & \text{für } x < 0 \end{cases}$$

Diese Funktion hat folgendes Schaubild:

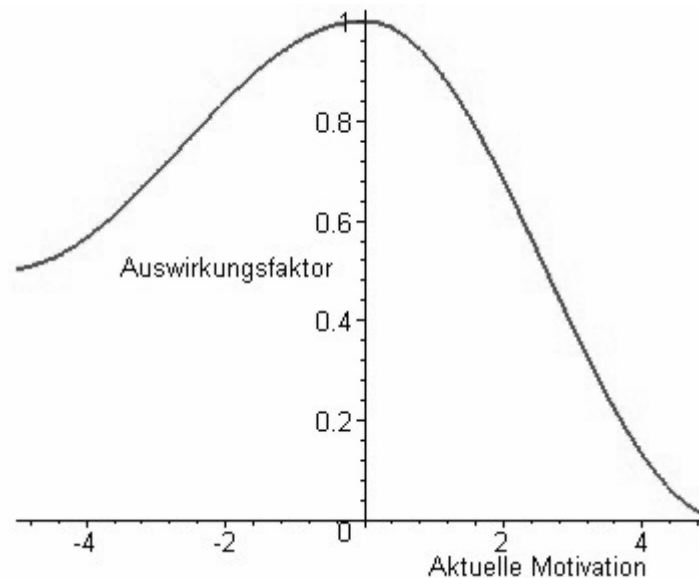


Abbildung 4: Anpassungsfunktion für die Motivation bei positiven Änderungen

An diesem Schaubild läßt sich dieser Effekt sehr gut anhand eines Beispiels erklären. Angenommen ein Ereignis hat eine motivationssteigernde Wirkung von +2. Ist der Mitarbeiter durchschnittlich motiviert (Motivation = 0), so steigt seine Motivation auch tatsächlich um den Wert 2 (Basisänderungswert). Je höher die Motivation des Mitarbeiters ist, desto weniger steigert das Ereignis seine Motivation. Ist der Mitarbeiter beispielsweise hochmotiviert (Motivation = 3), so wirkt sich das o.g. Ereignis nur noch mit einem Wert von ca. +0.8 auf die Motivation aus. Ähnlich verhält es sich im negativen Bereich. Hier wurde die Kurve jedoch abgeflacht, um zu vermeiden, daß Mitarbeiter, die eine schlechte Motivation besitzen nie wieder aus dem Motivationskeller herauskommen können.

Für Ereignisse mit negativer Auswirkung auf die Motivation wird dieses Schaubild umgekehrt, so daß die Funktion für alle abgeflacht ist. Folgende Funktion ergibt sich:

$$\text{Motivationsanpassungsfaktor} = \begin{cases} 0.5 * (\cos(0.6*x) + 1) & \text{für } x \leq 0 \\ 0.25 * (\cos(0.6*x) + 3) & \text{für } x > 0 \end{cases}$$

Diese Funktion besitzt folgendes Schaubild:

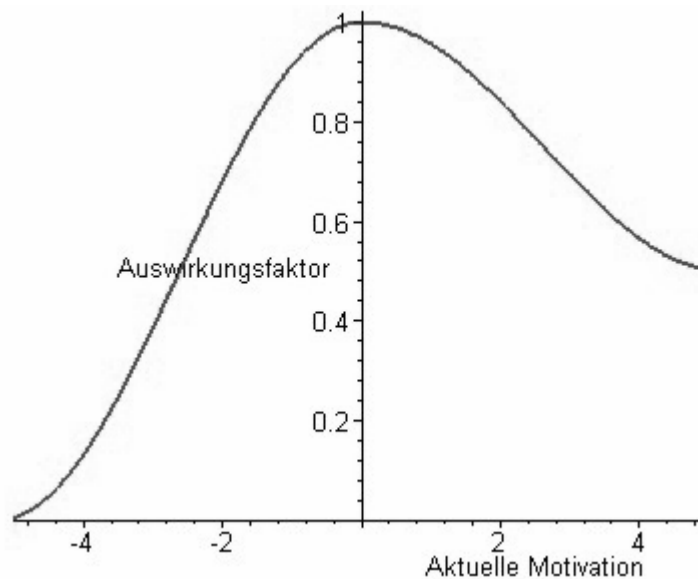


Abbildung 5: Anpassungsfunktion für die Motivation bei negativen Änderungen

4.2 Art der Tätigkeit

Es wurde gezeigt, daß die Motivation eines Entwicklers in großem Maße davon abhängt, wie hoch seine Erfahrung in dieser Tätigkeit ist. Dieser Effekt verstärkt sich für Mitarbeiter in der EDV-Branche sogar noch. Hier wird davon ausgegangen, daß eine mittlere Erfahrung etwa dazu beitragen sollte, daß die Motivation trotz des o.g. "Vergessenseffektes" konstant bleibt. Die Motivationänderung für die mittlere Erfahrungsstufe muß also leicht positiv sein.

Berechnungsvorschriften

- Ist ein Mitarbeiter mit einer Tätigkeit beschäftigt, mit der seine Erfahrung "KEINE" oder "GERING" ist, so vermindert sich die Motivation. Ist ein Mitarbeiter mit einer Tätigkeit beschäftigt, mit der seine Erfahrung "MITTEL" oder "HOCH" ist, so steigt die Motivation.

$$\text{Motivation} = \text{Motivation} \pm f(\text{Erfahrung in der aktuellen Tätigkeit})$$

Auch hier bezieht sich die Berechnung des Wertes auf eine Veränderung der Motivation pro Monat. Ist ein Mitarbeiter zwei Monate mit einer Tätigkeit beschäftigt, die ihn nicht interessiert, in der er also nach der in Kapitel 3.2.1 beschriebenen Verkürzung keine Erfahrung hat, so sinkt seine Motivation von sehr motiviert auf durchschnittlich. Dies bedeutet eine Senkung des Motivationswertes von 3 (hochmotiviert) auf 0 (durchschnittlich) in ca. 42 Arbeitstagen, also eine Senkung von 0.07 pro Tag. Hierbei handelt es sich um den Basisänderungswert. Im Spiel wird dieser Faktor aber durch die Anpassungsfunktion der Motivation (s. Kap. 4.1) normalerweise verkleinert. Dies würde dazu führen, daß der Faktor fast keine Auswirkung mehr hat. Aus diesem Grund wird der Faktor für diese Quantifizierung verdoppelt, also auf 0.14 pro Tag gesetzt, um den gewünschten Effekt zu erzielen.

Da es sich bei diesem Aspekt um den Faktor Arbeitsinhalt nach dem Herzbergschen Modell handelt, der einen Motivationsfaktor darstellt, erhöht sich der Wert nicht in dem Maße, wie er absinkt. Die Steigerung der Motivation wird auf ein Viertel gekürzt, so daß sich die Motivation in jedem Arbeitsschritt um bis zu 0.035 steigern kann. Eine genaue Übersicht über die Motivationssteigerung in Abhängigkeit von der Erfahrung in der aktuellen Tätigkeit ist in folgender Tabelle beschrieben

Erfahrung in der aktuellen Tätigkeit	Motivationsveränderung pro Arbeitstag
KEINE	-0.14
GERING	-0.07
MITTEL	0.0175
HOCH	0.035

Tabelle 5: Motivationsänderung nach Erfahrungsklassen

4.3 Mitarbeitergespräch

Ein Mitarbeitergespräch dient nicht nur der Information des Projektleiters über die Motivation des Mitarbeiters, sondern steigert auch aufgrund der höheren Anerkennung die Motivation des Mitarbeiters.

Berechnungsvorschriften

- Ein durchgeführtes Mitarbeitergespräch erhöht die Motivation:

$$\text{Motivation} = \text{Motivation} + x$$

Ein Mitarbeitergespräch steigert die Motivation des Mitarbeiters um 1.5 und kann nur maximal alle zwei Wochen durchgeführt werden.

Sinkt die Motivation eines Mitarbeiters unter -2, so fordert er seinen Projektleiter in 50% aller Fälle zu einem Mitarbeitergespräch auf. Auch diese Mitarbeitergespräche werden, selbst bei andauernd niedriger Motivation, nur maximal alle zwei Wochen durchgeführt.

4.4 Veränderliche Erfahrungen

Die Erfahrung wird im QS-Modell als variabler Wert definiert, der sich mit der Zeit und je nach Tätigkeit des Entwicklers verändert.

Bisher sind für die Erfahrung der Mitarbeiter folgende Stufen definiert [Drapp00]:

- *keine*: Ein Entwickler hat weder theoretische Kenntnisse, etwa durch Schulungen oder Vorlesungen, noch praktische Erfahrungen in einer Tätigkeit.
- *gering*: Ein Entwickler hat in einer Tätigkeit theoretische Kenntnisse oder praktische Erfahrung in einem Softwareprojekt erworben.
- *mittel*: Ein Entwickler hat eine Tätigkeit mindestens ein halbes Jahr lang ausgeübt.
- *hoch*: Ein Entwickler hat eine Tätigkeit mindestens ein Jahr lang ausgeübt.

Die Realisierung soll die Erfahrungen anhand der durchgeführten Tätigkeiten erhöhen. Hierfür wird der Wert *Erfahrungstage* eingeführt, der pro Tag um eins erhöht wird. Er

bezeichnet also die Anzahl der Tage, die sich ein Mitarbeiter mit einer bestimmten Tätigkeit beschäftigt hat. Diesem Wert zufolge wird dann die Erfahrungsstufe (im Modell: “*Erfahrung ...*”) angepaßt. Die Erfahrungstage lassen sich dann wie folgt in die Stufen umrechnen:

- *keine*: <2 Monate (< 60 Tage)
- *gering*: 2-6 Monate (61 - 180 Tage)
- *mittel*: 6-12 Monate (181 Tage - 360 Tage)
- *hoch*: >12 Monate (> 360 Tage)

Dies bedeutet, daß die Erfahrung nicht linear ansteigt, sondern am Anfang schneller zunimmt.

Die Projekterfahrung ist die einzige Erfahrung, die auch abnehmen kann. Ist der Mitarbeiter nicht im Projekt beschäftigt, so nimmt der Wert “*Erfahrungstage im Projekt*” mit jedem Tag um eins ab. Dies läßt sich in der Art erklären, daß in der gleichen Rate, in der sich eine Projekterfahrung aufbaut, sich auch die Prozesse innerhalb eines Projektes ändern. Also nimmt die Projekterfahrung im gleichen Maße ab, wie sie zunimmt.

4.5 Überstunden

Überstunden wurden neu in das Modell eingefügt. Sie werden durch zwei Tatsachen gekennzeichnet:

- Eine gewisse Anzahl von Überstunden hat keine Auswirkung auf die Motivation.
- Nach einer Phase von Überstunden benötigt der Mitarbeiter die doppelte Zeit um sich davon “zu erholen”

Hierfür wurde für jeden Entwickler ein Wert eingeführt, der ein Überstundenkonto repräsentiert. Auf diesem Konto kommen Überstunden hinzu, die angeordnet wurden. Dieses Konto wird in der überstundenfreien Zeit langsam abgebaut. Überschreitet das Konto einen bestimmten Wert, so hat dies negative Auswirkungen auf die Motivation.

Berechnungsvorschriften

- Überstunden verringern die Motivation:

$$\text{Motivation} = \text{Motivation} - \text{Überlastungswert}$$

Der Mitarbeiter besitzt die Möglichkeit, durch ein Kommando einem Entwickler über einen bestimmten Zeitraum zwei Überstunden täglich anzuordnen. Dies entspricht einer Mehrarbeitszeit von 25%. Die Anzahl der Überstunden in der Entität Entwickler wird also an jedem Überstundentag um 2 erhöht. Da der Mitarbeiter die doppelte Zeit benötigt, um sich zu erholen, wird der Wert also jeden Tag um eins gesenkt wenn der Mitarbeiter keine Überstunden mehr leistet.

Liegt der Überstundenwert über 20, was einer Überstundenphase von mehr als 2 Wochen entspricht, so wird dies als negatives Erlebnis für den Mitarbeiter gewertet und seine Motivation wird gesenkt.

Zur Quantifizierung des Ausmaßes um das die Motivation gesenkt wird, wird angenommen, daß eine Belastung durch Überstunden (Belastung = Wert >20) über einen Zeitraum von 4 Wochen die Motivation eines Mitarbeiters von hochmotiviert auf durchschnittlich

motiviert senkt. Dies entspricht einer Senkung der Motivation um 3 Motivationspunkte in 20 Tagen, also um 0.15 Motivationspunkte pro Tag.

4.6 Unterbeschäftigung

Die Unterbeschäftigung berechnet sich analog zur Überlastung. Auch hier wird der Wert der Stunden, die der Entwickler nicht beschäftigt war, in einem "Unterlastkonto" gespeichert. Anders als bei Überstunden sinkt dieser Wert aber nicht erst bei Beendigung der Unterforderung, sondern der Mitarbeiter beginnt sofort damit, die Unterlast zu vergessen. Dies hat den Grund, daß etwa eine Stunden ohne Beschäftigung pro Tag als folgenlos angesehen wird. Der Mitarbeiter baut, analog zu den Überstunden, eine Stunde pro Tag ab. Steigt das Konto über 20, was entweder etwa 3 Tagen ohne Beschäftigung oder zwei Wochen mit einer täglichen Unterbeschäftigung von 3 Stunden entspricht, so sinkt die Motivation analog zur Überlast um 0.15 Punkte pro Tag.

4.7 Motivationsmaßnahmen

Motivationsmaßnahmen werden neu eingeführt als Mittel um die Motivation der Entwickler anzuheben. Es stehen verschiedene Motivationsmaßnahmen zur Auswahl, die verschiedene Motivationssteigerungen zur Folge haben.

Berechnungsvorschriften

- Teambildende Maßnahmen erhöhen die Motivation:

$$\text{Motivation} = \text{Motivation} + f(\text{Art der Maßnahme})$$

Eine Motivationsmaßnahme hat folgende Auswirkungen auf die Motivation der teilnehmenden Mitarbeiter:

Motivationsmaßnahme	Motivationsänderung
Kuchenessen	+ 0.7
Mittagessen	+ 0.9
Abendessen	+ 1.1
Motivationsseminar 1/2 Tag	+ 2
Motivationsseminar 1 Tag	+ 3.5

Tabelle 6: Motivationsänderung je Motivationsmaßnahme

Um ein Hochpushen der Motivation zu vermeiden, kann nur eine Motivationsmaßnahme pro Woche durchgeführt werden.

Für Motivationsmaßnahmen werden folgende Kosten zu den Projektkosten addiert und die Mitarbeiter haben folgenden Arbeitsausfall:

Gemeinsames Kuchenessen am Nachmittag (5 DM/Person + 1h Arbeitsausfall)

Gemeinsames Mittagessen (30 DM/Person + 3h Arbeitsausfall)

Gemeinsames Abendessen (40 DM/Person + 1h Arbeitsausfall)

Motivationsseminar 1/2 Tag (400 DM/Person + 4h Arbeitsausfall)

Motivationsseminar 1 Tag (800 DM/Person + 8h Arbeitsausfall)

4.8 Produktivität

In [Drapp00] sind die bisherigen Produktivitätswerte abhängig von der Erfahrung und der Teamgröße modelliert.

In [Jones86] wird folgende Studie zur Erfassung der Abhängigkeit der Produktivität von der Motivation durchgeführt:

Es wurden zwei Projekte miteinander verglichen, die sich nur durch die Motivation der Mitarbeiter unterschieden. In der Studie wird dies dadurch reflektiert, daß in einem Projekt angemessen bezahlte und gute Mitarbeiter eingesetzt werden, die in einer richtigen Zusammensetzung in das Projekt gesetzt wurden. In einem zweiten Projekt werden schlecht bezahlte und nur durchschnittliche Mitarbeiter eingesetzt, die des weiteren noch wild zusammengewürfelt sind, während die guten Mitarbeiter zu Managern befördert wurden. Jones stellt in diesen Projekten für die gleiche Aufgabe Aufwandsunterschiede von 29% (hochmotiviert: 1085 MM, unmotiviert: 1400 MM) fest. Die bisherige Berechnung der Produktivität, abhängig von der Erfahrung und Teamgröße, wird hier nicht geändert. Es wird nur die Berechnungsvorschrift für die Quantifizierung wie folgt verändert:

Berechnungsvorschriften

- Für alle Produktivitätsfaktoren gilt:

$$\text{Produktivität} = f(\text{Erfahrung}, \text{Teamgröße}, \text{Motivation})$$

Für die Quantifizierung wird nur der von der Motivation abhängige Teil betrachtet.

Die in [Jones86] ermittelten Werte sind für das QS-Modell nur teilweise aussagekräftig, da Jones nur wenige Motivationsfaktoren berücksichtigt, die dem Modell zugrunde liegen (Jones: Bezahlung, Teamwork). Wichtige Faktoren wie beispielsweise die Art der Tätigkeit, die Anerkennung oder die Mehr- oder Minderarbeit sind nicht berücksichtigt. Ich gehe deshalb davon aus, daß sich die Unterschiede zwischen einem hochmotivierten und einem sehr unmotivierten Mitarbeiter eher verstärken. Dies ist auch der Eindruck, der einem aus der Praxis heraus entsteht. Der Gesamtunterschied zwischen einem sehr unmotivierten Mitarbeiter und einem hochmotivierten Mitarbeiter beträgt im erweiterten QS-Modell 50%. Ein hochmotivierter Mitarbeiter besitzt also eine um 50% höhere Produktivität als ein sehr unmotivierter Mitarbeiter (Produktivität sehr schlechter Motivation * 1.5 = Produktivität sehr hoher Motivation).

Da zur Quantifizierung nur sehr wenig Daten vorliegen (zwei!), läßt sich eigentlich nur eine lineare Funktion errechnen. In der Realität ist dies jedoch nicht der Fall. Ich gehe davon aus, daß sich die Auswirkung der Motivation auf die Produktivität in Extrembereichen abschwächt. Es wird also eine Funktion benötigt, die im Bereich um den Standardwert stärker ansteigt als in Extrembereichen. Für diesen Zweck eignet sich ein Arcustangens sehr gut. Dieser ist auch mit einfachen arithmetischen Mitteln zu berechnen. Der Arcustangens soll die Produktivitätssteigerung bzw. -senkung ausdrücken. Er muß also mit Hilfe von Faktoren in das benötigte Schema gepreßt werden. Um einen Anstieg in

der Produktivität von insgesamt 50% (s.o.) zu erreichen, muß ein Bereich von 20% um einen Standardwert gefunden werden. Da der Standardwert hier eins ist (100% Produktivität), schwankt der Bereich der Produktivitätsänderung zwischen -20% und +20%, also einem Multiplikationsfaktor von 0.8 bis 1.2. Der Arcustangens muß folgende Werte liefern:

Motivation	Motivation (Zahlenwert)	Produktivitätsänderung
sehr hoch motiviert	+5	* 1.2
ausgeglichen	0	0* 1
sehr schlecht motiviert	-5	* 0.8

Tabelle 7: Beispiele für Produktivitätsänderungen je nach Motivation

Daraus ergibt sich folgende Funktion,

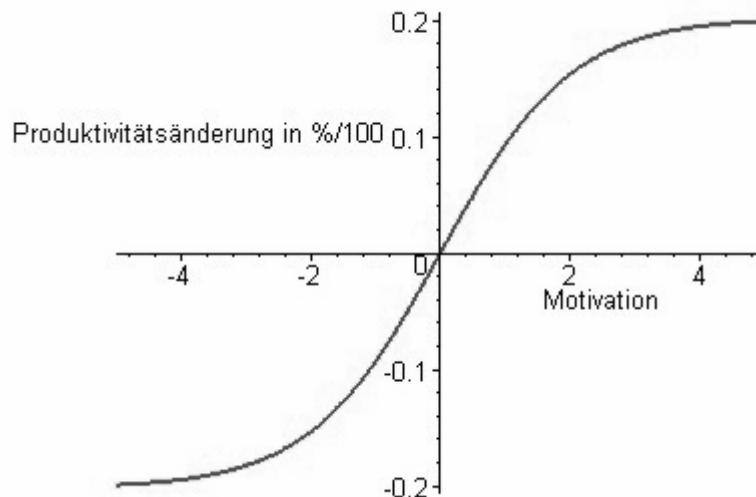


Abbildung 6: Anpassungsfunktion der Produktivität in Abhängigkeit zur Motivation

die sich wie folgt darstellt:

$$PA(\text{Motivation}) = 0.202 * \tanh(0.5 * \text{Motivation})$$

Diese Funktion ist für das Modell aufgrund des Definitionsbereiches der Motivation nur zwischen +5 und -5 definiert und hat für diesen Definitionsbereich den Wertebereich [-0.2;+0.2].

Die Gesamtformel für die Produktivität eines Mitarbeiters lautet also:

$$\text{Produktivität} = f(\text{Erfahrung}, \text{Teamgröße}) * [1 + 0.202 * \tanh(0.5 * \text{Motivation})]$$

4.9 Fehlzeiten

In [AOK98] sind die durchschnittlichen Fehlzeiten in der deutschen Wirtschaft angegeben. Diese Zahlen sind nach Berufsgruppen unterteilt, wobei für das QS-Modell die Berufsgruppe der Ingenieure verwendet wird. Diese Daten wurden jedoch nur aufgrund von AOK-Daten erfaßt. Hierbei ist anzumerken, daß die AOK im Bereich von Ingenieuren oder EDV-Spezialisten sicherlich nicht die größte Krankenkasse ist. Die Daten können aber aufgrund der dennoch hohen Anzahl von Ingenieuren in der AOK als repräsentativ angesehen werden.

Die AOK gibt einen Prozentsatz von 2% Fehlzeit für die bei ihr versicherten Ingenieure an.

Die Abhängigkeit der Fehlzeit von der Motivation wird in [Couga78] beschrieben. Hier wird gezeigt, daß über alle Berufsgruppen hinweg durch eine hohe Motivation eine Senkung der Fehlzeiten von 7 auf 3 Tage möglich ist, was einer Steigung und einer Senkung um 40% um einen Mittelwert entspricht.

Berechnungsvorschriften

- Die Wahrscheinlichkeit, daß ein Mitarbeiter nicht zur Arbeit kommt, hängt von seiner Motivation und dem Zufall ab.

Die persönliche Fehlzeitwahrscheinlichkeit wird für jeden Mitarbeiter separat festgelegt und als F_p bezeichnet. Die durchschnittliche Wahrscheinlichkeit, daß ein Mitarbeiter an einem Tag krank wird, beträgt laut [AOK98] im Schnitt 2%. Dies wird als Standardwert eingesetzt, kann aber durch das Situationsmodell verändert werden, um verschiedene Fehlzeitenwahrscheinlichkeiten verschiedener Mitarbeiter realisieren zu können. Es wird jedoch empfohlen, diesen Wert nicht zu sehr zu verändern, da eine zu hohe Streuung der Werte die Abhängigkeit von der Motivation undurchschaubar machen könnte.

Angenommen wird, daß sich die Fehlzeiten zwischen den motivationsbedingten Höchst- und Niedrigstwerten linear verhalten und der Mittelwert der durchschnittlichen Fehlzeit entspricht. Angepaßt auf die Motivation ergibt sich eine Spanne von 40% um diesen Wert. Werden wiederum die Werte +5 und -5 für eine besonders gute und eine besonders schlechte Motivation festgelegt, so ergibt sich folgende Funktion, die anhand der Motivation die Wahrscheinlichkeit einer Fehlzeit berechnet:

$$\text{Fehlzeitwahrscheinlichkeit } F = (-0,08 * M + 1) * F_p$$

Auch diese Funktion ist durch den Definitionsbereich der Motivation ([-5;5]) nach oben und unten auf $F_p * 1,4$ und $F_p * 0,6$ beschränkt.

Dieser Wert gibt die Wahrscheinlichkeit an, daß ein Mitarbeiter an einem bestimmten Tag genau einen Tag lang krank ist. Da jedoch auch mehrtägige Krankheiten im Modell enthalten sein müssen, wird die Wahrscheinlichkeit hierfür angepaßt. Eine fünftägige Krankheit

darf nicht mit der Wahrscheinlichkeit für eine eintägige Krankheit auftreten, sondern muß fünfmal so niedrig sein.

Länge der Krankheit	Wahrscheinlichkeit
1 Tag	F
2 Tage	F * 0,5
5 Tage (= 1 Woche)	F * 0.2

Tabelle 8: Wahrscheinlichkeiten für die Länge der Fehlzeiten

Diese Tabelle beinhaltet die Wahrscheinlichkeiten, daß ein Mitarbeiter an einem bestimmten Tag für einen bestimmten Zeitraum krank wird, also auch die Wahrscheinlichkeiten, mit denen die entsprechenden Regeln feuern müssen.

4.10 Kündigung durch schlechte Motivation

Als letztes Mittel im Rahmen einer sehr niedrigen Motivation sieht der Mitarbeiter nur die Kündigung. Fällt die Motivation also unter einen bestimmten Schwellenwert, so reicht er seine Kündigung ein. Durch das Ausscheiden eines Mitarbeiters wird das Teamwork verändert. Dadurch sinkt die Motivation der übrigen Mitarbeiter.

Ein Mitarbeiter sucht ab einer Motivation von -2 den Projektleiter in 50% der Fälle zu Mitarbeitergesprächen auf. Ändert sich an seiner Situation nichts und verschlechtert sich die Motivation weiter, so wird er bei einer Motivation von -4,5 seine Kündigung einreichen. Die Kündigung ist unwiderrufbar. Nach einer Kündigungsfrist von 1 Monat wird der Mitarbeiter das Projekt verlassen. Durch die Kündigung eines Mitarbeiters verändert sich auch das Teamwork der anderen Mitarbeiter. Dieser Effekt wird in Kapitel 4.11 beschrieben.

4.11 Unvorhersehbare Kündigung

Um der Realität gerecht zu werden, wurden in das erweiterte QS-Modell zufällige Kündigungen aufgenommen, die der Spieler nicht beeinflussen oder voraussehen kann.

Berechnungsvorschriften

- Zufällige, unvorhersehbare Ereignisse bewegen einen Mitarbeiter dazu, sofort die Kündigung einzureichen.
- Das Ausscheiden eines Mitarbeiters hat Auswirkungen auf die Motivation der anderen Mitarbeiter.

Eine unvorhersehbare Kündigung kommt in der Realität wahrscheinlich in einem Projekt mit ca. 10 Mitarbeitern etwa jedes Jahr einmal vor. Dies entspricht einer Kündigungswahrscheinlichkeit von $1 / (225 \text{ Tage} * 10 \text{ Mitarbeiter}) = 0,044\%$ pro Tag. Um den Effekt aber in SESAM realisieren zu können, muß er etwas übertrieben werden. Ich setze den Wert daher standardmäßig auf 0.1%, was etwa einer Kündigungsrate von 1 von 10 Mitarbeitern pro Halbjahr entspricht. Je nach Ausbildungsziel und Schwierigkeitsgrad des jeweiligen Experimentes sollte dieser Wert angepaßt werden.

Des weiteren wird durch das Ausscheiden eines Mitarbeiters das Teamwork und damit die Motivation des gesamten Teams verändert. Hatte der Mitarbeiter eine hohe Motivation, so senkt sich die Motivation der übrigen Mitarbeiter, war die Motivation des Ausscheidenden niedrig, so steigt die Motivation der übrigen, da hier ein "Mitläufer" das Projekt verlassen hat. Folgende Auswirkungen ergeben sich:

Motivation M des Ausscheidenden (vgl. Kap. 8)	Motivationsänderung der übrigen Mitarbeiter
sehr hoch	- 1.0
hoch	- 0.5
niedrig	+ 0.5
sehr niedrig	+ 1.0

Tabelle 9: Motivationsänderung der anderen Mitarbeiter nach einer Kündigung

Hat ein Mitarbeiter eine Projekterfahrung, die größer ist als 2 Monate (45 Tage), so verdoppeln sich die Auswirkungen.

Die Quantifizierung mag hier im Vergleich zu anderen Aspekten sehr hoch dimensioniert sein. Dies hat zwei Gründe: Erstens soll der Spieler lernen, seine Mitarbeiter zu beobachten und schlecht motivierte Mitarbeiter aus dem Projekt zu entfernen, zweitens ist es auch in der Realität der Fall, daß durch das Verlassen eines sehr hoch motivierten Mitarbeiters, der viel Projekterfahrung besitzt, die Motivation der übriggebliebenen Teammitglieder sehr stark sinkt.

4.12 Projektstatus

Die Berechnung eines Projektstatus ist nicht trivial. Die Frage wie es um ein Projekt steht, kann oft nur mit Kenntnis aller Umgebungseffekte beantwortet werden, die aber in einem Computermodell nicht zur Verfügung stehen. Hier wird nun versucht, die fehlenden Kenntnisse aus den in SESAM vorhandenen Werten abzuleiten.

Für die Berechnung des Projektstatus müssen drei Fragen beantwortet werden:

- In welcher Phase sollte sich das Projekt momentan befinden (Spezifikation, Entwurf, Codierung/Modultest, Integration/Test) ?
- Welche Aufwände hat ein durchschnittliches vergleichbares Projekt bis zum aktuellen Zeitraum verbraucht ?
- Welche Aufwände sind im aktuellen Projekt bisher aufgelaufen ?

Werden die Antworten auf diese Fragen miteinander verglichen, so kann eine Aussage über den Status des Projektes gegeben werden.

Aktuelle Projektphase und durchschnittliche Aufwände

In den Untersuchungen von Boehm wurden Zeitannteile und Aufwände festgelegt, die ein Projekt für die verschiedenen Phasen aufbringen sollte::

Phase	Aufwand in %	Kumulierter Aufwand in %	Zeit in %	Kumulierte Zeit in %
Spezifikation	16	16	19	19
Entwurf	25	41	23	42
Codierung + Modultest	40	81	35	78
Integration + Test	19	100	22	100

Tabelle 10: Theoretischer Aufwand und Zeitbedarf für einzelne Projektphasen [Boehm86]

Für die typische Klasse der Projekte in SESAM wird die Boehm'sche Klasse Organic und Intermediate (8KDSI) verwendet, da diese Klasse den in SESAM modellierten Projekten am nächsten kommen. Eine Differenzierung bezüglich der Größe des Projektes wird aufgrund der sehr geringen Änderungen in den Werten Boehms nicht vorgenommen, sondern die Projektklasse Intermediate wird als Durchschnitt verwendet

Durch einen einfachen Vergleich der bisherigen Projektdauer mit der Gesamtlaufzeit des Projektes läßt sich nun bestimmen, in welcher Phase sich das Projekt laut Boehm befinden sollte. Alle Dokumente, die bis zu diesem Zeitpunkt fertiggestellt sein müßten, werden zur Berechnung des Projektstatus herangezogen.

Zur Veranschaulichung wird diese Berechnung anhand eines Beispielen verdeutlicht: Ein Projekt läuft über einen Gesamtzeitraum von 250 Tagen. Nach 75 Tagen soll der Projektstatus berechnet werden. Laut Tabelle 10 ist nach 30% der verbrauchten Zeit nur die Spezifikationsphase beendet. Es wird also nur die Spezifikation in die Berechnung einbezogen. Alle anderen Dokumente sind zu diesem Zeitpunkt noch in Arbeit oder noch nicht beendet. Also wäre in einem fiktiven Statusmeeting über diese Dokumente auch noch keine Rechenschaft abzulegen.

Aktueller Fortschritt des Projektes

Der Projektfortschritt wird aus drei Werten berechnet:

1. Aufwandsabweichung des Projektes zu durchschnittlichen Projekten:

Aufwandsabweichung =

$$- \left(\left[\frac{\text{Aufwand je Phase}}{\text{Theoret. Aufwand je Phase}} \right] - 1 \right)$$

Mit dieser Formel wird für jede bisherige Projektphase ein Wert berechnet, der die Abweichung zwischen dem wirklichen Aufwand für diese Phase und dem durch Boehm vorgegebenen Aufwand für diese Phase zeigt.

Die Aufwandsabweichung wird als Real im Wertebereich [-1 ; inf] modelliert und für alle bisher durchgeführten Phasen berechnet.

Aus allen Aufwandsabweichungen wird der Durchschnitt gebildet, der die Gesamtaufwandsabweichung über alle Phasen darstellt.

Auch hier wird wieder auf das o.g. Beispiel zurückgegriffen. In einem ersten Schritt muß der theoretische Aufwand für die Spezifikationsphase berechnet werden. Es handelt sich um ein Projekt, das 200 AFP umfasst und in Ada95 programmiert wird. Die Formel, die Boehm für die gewählte Projektklasse liefert, lautet $MM = 2.4 (KDSI)^{1,05}$. Alle Werte eingesetzt ergibt sich ein Gesamtaufwand von ca. 26 Mannmonaten (MM). Anteilmäßig ergibt sich ein Aufwand für die Spezifikation von 4.16 MM. Laut Projekthandbuch wurden für die Spezifikation aber nur 3.5 MM aufgewendet. Dies entspricht einer Aufwandsabweichung von 0.16. Es wurde also 16% weniger Aufwand benötigt.

2. Vollständigkeitsgrad aller bisherigen Dokumente

Der Vollständigkeitsgrad der Dokumente wird aus der Anzahl der Funktion-Points berechnet, die in den Dokumenten enthalten sind, die bisher hätten fertiggestellt werden müssen. Anhand der bisher in den Dokumenten enthaltenen AFP wird die Abweichung zu der vorgeschriebenen Anzahl AFP (festgelegt durch die Aufgabenstellung) berechnet.

Vollständigkeitsgrad je Dokument =

$$[(\text{Aktuelle Anzahl AFP}) / (\text{Vorgeschriebene Anzahl AFP})] - 1$$

Durch die Orientierung an der Aufgabenstellung ist es möglich, eine höhere Vollständigkeit zu erreichen, als es nötig ist. Diese Tatsache fließt also positiv in den Projektstatus ein. Für Dokumente, für die in der Aufgabenstellung keine Vollständigkeitswerte angegeben werden, wird der Vollständigkeitsgrad des Codes verwendet. Der Gesamtwert der Vollständigkeit ergibt sich aus den Mittelwerten aller Werte für die einzelnen Dokumente und liegt im Bereich zwischen unendlich und -1.

Auch dies wird wieder am Beispiel verdeutlicht:

Für die Spezifikation wird in der Aufgabenstellung kein Vollständigkeitsgrad vorgegeben, also wird der Vollständigkeitsgrad vom Code verwendet. Er beträgt 95%. In diesem Fall müssen also 180 AFP im Code enthalten sein. Laut Spezifikation befinden sich aber nur 170 AFP im Code. Es ergibt sich also ein Vollständigkeitsgrad von -0.058. Gegenüber dem Referenzwert fehlen also 5,8 % der AFP.

3. Qualitätsgrad der Dokumente

Den Qualitätsgrad eines Dokumentes feststellen zu können ist eine nicht gerade triviale Aufgabe. In einem Simulationsmodell, in dem keine persönlichen Aspekte in die Berechnung einfließen können ist dies nochmals schwerer. Aus diesem Grund wird hier eine Verkürzung vorgenommen. Die Qualität eines Dokumentes wird hier mit der Fehlerrate im Dokument gleichgesetzt. Die Fehler im Dokument sind die einzige Möglichkeit innerhalb des QS-Modells auf Qualitätsaspekte zurückzugreifen.

Der Qualitätsgrad berechnet sich aus dem Wert der aktuell im Dokument enthaltenen Fehler, geteilt durch den Wert an Fehlern, die laut Aufgabenstellung im Dokument enthalten sein dürfen.

Qualitätsgrad je Dokument =

$$[(\text{Erlaubte Anzahl Fehler}) / (\text{Enthaltene Anzahl Fehler})] - 1$$

Sind für einzelne Dokumente keine Fehlerraten vorhanden, so wird die Fehlerrate des Codes verwendet. Auch hier wird analog zur Vollständigkeitsrate der Mittelwert aller Dokumente gebildet. Dieser liefert den Wert des Gesamtqualitätsgrades.

Auch hier wieder zum Beispiel:

Im Code sind 5 Fehler/KLOC erlaubt. Die Spezifikation enthält insgesamt 53 Fehler. Hier müssen zuerst die erlaubten Fehler im Dokument ausgerechnet werden. Da in Ada95 einem AFP etwa 49 LOC entsprechen kommt man auf einen Wert von 0.245 Fehler/AFP, also einer Gesamtfehlerzahl von 49 Fehlern in der Spezifikation. Dies entspricht einem Qualitätsgrad von -0.07 . Das Dokument enthält 7% mehr Fehler als erlaubt.

Projektstatus aus Sicht des Kunden

Für die endgültige Berechnung des Projektstatus aus Sicht des Kunden werden die Aufwandsabweichung, der Gesamtvollständigkeitsgrad und die Gesamtqualität herangezogen. Da ein schlechter Wert in einen Teilbereich zum Teil durch einen guten Wert im anderen Teilbereich ausgeglichen werden kann, werden die drei Werte zuerst gewichtet und dann gemittelt. Durch diese Gewichtung werden positive Werte durch vier geteilt, um ein Ausgleichen von schlechten Werten durch im gleichen Maße gute Werte zu verhindern. Dies entspricht auch der Realität, da schlechte Ergebnisse in einigen Teilbereichen nur in geringem Maße durch gute Ergebnisse in anderen Teilbereichen ausgeglichen werden können. So bringt es beispielsweise nichts, wenn ein Projekt 25% weniger Zeit verbraucht, aber dafür 25% mehr Fehler beinhaltet. Das Projekt wird letztendlich negativ bewertet werden. Diese Tatsache wird mit der Gewichtung berücksichtigt.

Der Projektstatus wird also wie folgt berechnet:

$$\text{Projektstatus} = \frac{(\text{Gesamtaufwandsabweichung}_{\text{gewichtet}} + \text{Gesamtvollständigkeitsgrad}_{\text{gewichtet}} + \text{Gesamtqualitätsgrad}_{\text{gewichtet}})}{3}$$

Im obigen Beispiel würde sich der Projektstatus aus Sicht des Kunden wie folgt berechnen:

$$[(0.15 / 4) + (-0.058) + (-0.07)] / 3 = -0.0302$$

Der Projektstatus aus Sicht des Kunden wäre also negativ.

Projektstatus aus Sicht des Senior Management

Der Projektstatus des Senior Management unterscheidet sich marginal von dem des Kunden. Da das Senior Management normalerweise keine Informationen über die Qualität der erstellten Dokumente hat, wird der Qualitätsgrad nicht mit in die Berechnung einbezogen. Für die endgültige Berechnung des Projektstatus werden also die Aufwandsabweichung und der Gesamtvollständigkeitsgrad herangezogen. Der Status wird wie folgt berechnet:

$$\text{Projektstatus} = \frac{\text{Gesamtaufwandsabweichung}_{\text{gewichtet}} * \text{Gesamtvollständigkeitsgrad}_{\text{gewichtet}}}{2}$$

Der Projektstatus aus Sicht des Senior Managements würde sich nach dem Beispiel wie folgt berechnen:

$$[(0.15 / 4) + (-0.058)] / 2 = -0.01$$

Der Projektstatus ist also trotz des erheblich geringeren Aufwandes noch knapp im Minus.

4.13 Motivation des Senior Managements

Auch das Senior Management muß durch den Projektleiter beachtet werden. Das Verhältnis des Senior Managements zum Projekt ist im Modell durch die Motivation des Senior Managements repräsentiert. Es hat vor allem Auswirkungen auf die Häufigkeit der Statusmeetings.

Berechnungsvorschriften

- Die Motivation des Senior Managements berechnet sich während eines Statusmeetings aus dem Wert des Projektstatus.

$$\text{Motivation} = f(\text{Projektstatus}(\text{SeniorManagement}))$$

- Der Zeitabstand zwischen zwei Statusmeetings wird durch die Motivation des Senior Managements bestimmt.

Auch hier ist eine Quantifizierung schwierig, da es in der Literatur keine quantitativen Aussagen gibt. Die Motivation des Senior Managements wird während eines Statusmeetings auf den Wert des Projektstatus gesetzt. Diese Motivation beeinflusst dann die Abstände zwischen den Statusmeetings.

Es wird davon ausgegangen, daß ein normaler Abstand zwischen zwei Statusmeetings in etwa einen Monat beträgt. Gerät das Projekt in kritische Phasen, wird der Informationsbedarf des Senior Managements größer und die Abstände sinken:

Motivation M	Abstand zwischen Statusmeetings
$M \leq -0.5$	1 Woche (7 Tage)
$-0.1 < M < -0.5$	2 Wochen (14 Tage)
$M > -0.1$	1 Monat (30 Tage)

Tabelle 11: Auswirkung der Motivation des Senior Managements auf den Abstand zwischen zwei Statusmeetings

Sinkt die Motivation des Senior Managements unter 0.8, so wird der Projektleiter von seinen Pflichten entbunden, was im Spiel einem Projektabbruch gleich kommt.

4.14 Motivation des Kunden

Die Motivation des Kunden berechnet sich analog zu der des Senior Managements. Hier kommen jedoch zwei Punkte hinzu:

Berechnungsvorschriften:

- Die Motivation des Kunden berechnet sich während eines Statusmeetings aus dem Wert des Projektstatus.

$$\text{Motivation} = f(\text{Projektstatus}(\text{Kunde}))$$

- Der Zeitabstand zwischen zwei Statusmeetings wird durch die Motivation des Kunden bestimmt.

- Die Qualität der Dokumente in einem Review mit dem Kunden beeinflusst die Motivation des Kunden.
- Ein Review mit dem Kunden steigert die Motivation des Kunden.

Hinzu kommt also, daß sich die Motivation des Kunden auch durch ein Review verändert, da der Kunde auch hier einen Einblick in das Projekt bekommt. Reviews verhalten sich bezüglich der Kundenmotivation also gleich wie Statusmeetings. Zusätzlich ist der Kunde über eine Beteiligung an einem Review erfreut und so steigert sich seine Motivation nach jedem Review für vier Wochen um 0.5.

Die übrigen Auswirkungen des Statusmeetings mit einem Kunden (bzgl. Zeitabstände) verhalten sich analog zum Senior Management (s.o.).

Kapitel 5:

Test und Nachweis ausgewählter Aspekte

Wie die Spezifikation oder die Codierung ist auch der Test ein wichtiger Bestandteil eines jeden Softwareprojektes, so auch in dieser Arbeit.

Boehm unterscheidet zwei hauptsächliche Phasen des Tests, die Verifikation und die Validierung. Die Unterscheidung zwischen diesen Phasen beschreibt er einfach, aber passend mit zwei Fragen [Boehm86]:

Verification: "Are we building the product right?"

Validation: "Are we building the right product?"

Pressman bezeichnet die Phase des Modultests als Verifikation, also die Überprüfung ob das erstellte Produkt funktionstüchtig ist. Eine ebenfalls große Bedeutung hat die Validierung, also der Test, ob alle Anforderungen korrekt umgesetzt wurden.

In dieser Arbeit wird noch ein zusätzlicher Aspekt im Rahmen der Quantifizierung betrachtet. Da während der Arbeit eine nicht zu vernachlässigende Anzahl an Hypothesen aufgestellt wurde, muß nun festgestellt werden, ob das erstellte Gesamtprodukt plausible Ergebnisse liefert.

In diesem Kapitel werden einige ausgewählte Aspekte beschrieben die sowohl aus dem Bereich der Verifikation als auch der Plausibilitätsprüfung sind. Hierfür wurden Testfälle erstellt, die jeweils ausgewählte Aspekte einzeln oder im Zusammenhang darlegen sollen. Die Auswertung der Tests wurde vor allem mit dem für SESAM erstellten Auswertungstool SESAMALYZER durchgeführt.

5.1 Vergessen von Motivationseffekten

Sowohl positive als auch negative Ereignisse, die die Motivation betreffen, werden nach einer bestimmten Zeit vergessen (s. Kap. 4.1). In diesem Testfall sind drei Mitarbeiter in einem Projekt beschäftigt. Der erste Mitarbeiter startet hochmotiviert (Motivation = +3), der zweite durchschnittlich (Motivation = 0) und der dritte sehr unmotiviert (Motivation = -3). Die Mitarbeiter besitzen in ihrer Tätigkeit eine mittlere Erfahrung, um den Einfluß der Erfahrung auf die Motivation möglichst gering zu halten. Nach einer Zeit von 42 Tagen sollte sich die Motivation der Mitarbeiter an den Standardwert Null angenähert haben, da keine Motivationssteigernden oder -senkenden Ereignisse eintreten. Vollständig erreichen wird sie diesen Wert jedoch nicht, da die Motivation nicht linear absinkt. Bei Mitarbeiter 2 sollte sich die Motivation über das gesamte Projekt hinweg nicht oder nur sehr gering verändern.

Testauswertung

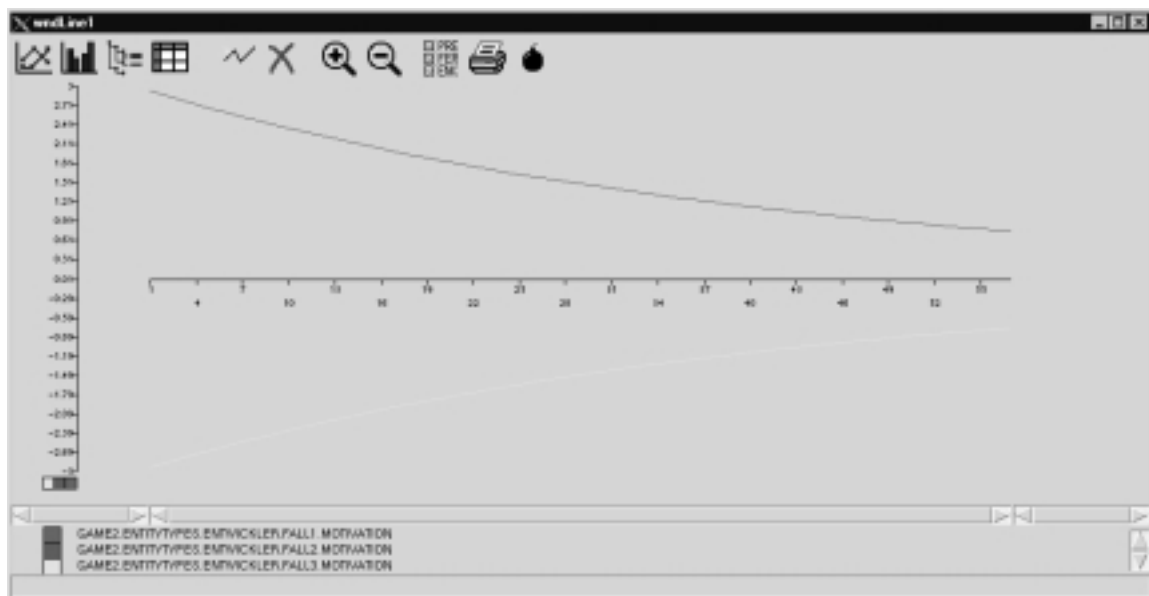


Abbildung 7: Testprotokoll Vergessen von Motivationsereignissen (Motivation)

Die Motivationen der Mitarbeiter verhielten sich erwartungsgemäß. Wie im Schaubild zu erkennen ist, nähert sich die Motivation ständig dem Standardwert an bzw. bleibt auf diesem. Die Motivation von Mitarbeiter 2 bleibt auf 0, während sich die Motivation der Mitarbeiter Nr. 1 und Nr. 3 auf 1 absenkte. Die Motivation des zweiten Mitarbeiters liegt im Bild genau unter der X-Achse.

5.2 Motivationsmaßnahmen heben die Motivation

Durch das Durchführen von Motivationsmaßnahmen läßt sich die Motivation von Mitarbeitern steigern. Verschiedene Motivationsmaßnahmen steigern die Motivation jedoch unterschiedlich (s. Kap. 4.7). Um diesen Effekt zu zeigen, werden zwei Tests durchgeführt. In jedem Testfall werden jeweils zwei Mitarbeiter beschäftigt. Einer der Mitarbeiter besitzt eine hohe Motivation (+3), der andere Mitarbeiter ist durchschnittlich motiviert

(0). Hierdurch läßt sich außerdem noch zeigen, daß sich gleiche Ereignisse nicht auf alle Mitarbeiter gleich auswirken. Je höher die Ausgangsmotivation eines Mitarbeiters ist, desto weniger Auswirkungen besitzt ein positives Ereignis (s. Kap. 4.1). Die einzelnen Testfälle laufen wie folgt ab:

Fall 1

Die Entwickler arbeiten an einer Tätigkeit, für die sie eine mittlere Erfahrung besitzen. Die Testdauer beträgt 28 Tage. Nach 14 Tagen wird eine Teamschulung durchgeführt.

Fall 2

Die Entwickler arbeiten an einer Tätigkeit, für die sie eine mittlere Erfahrung besitzen. Die Testdauer beträgt 28 Tage. Nach 14 Tagen wird eine Mittagessen durchgeführt.

Da die Mitarbeiter eine Tätigkeit ausüben, in der sie mittlere Erfahrung besitzen, wird die Motivation bei Mitarbeiter 1 aufgrund seiner hohen Motivation leicht sinken und bei Mitarbeiter 2 aufgrund der nur durchschnittlichen Motivation leicht steigen. In dem Moment, in dem die Motivationsmaßnahme durchgeführt wird, muß ein klarer Anstieg der Motivation sichtbar sein. Fall 1 muß die höchste Motivation liefern. In Fall 2 steigt die Motivation geringer an. Die gleiche Motivationsmaßnahme hat auf die verschieden motivierten Mitarbeiter verschiedene Auswirkungen. Beim hochmotivierten Mitarbeiter, der zum Zeitpunkt der Motivationsmaßnahme etwa eine Motivation von 2 besitzen sollte, wird der Effekt der Motivationsmaßnahme um etwa 40 % abgeschwächt. Der durchschnittlich motivierte Mitarbeiter sollte die Auswirkungen der Motivationsmaßnahme in vollem Maße erfahren.

Testauswertung

Fall 1

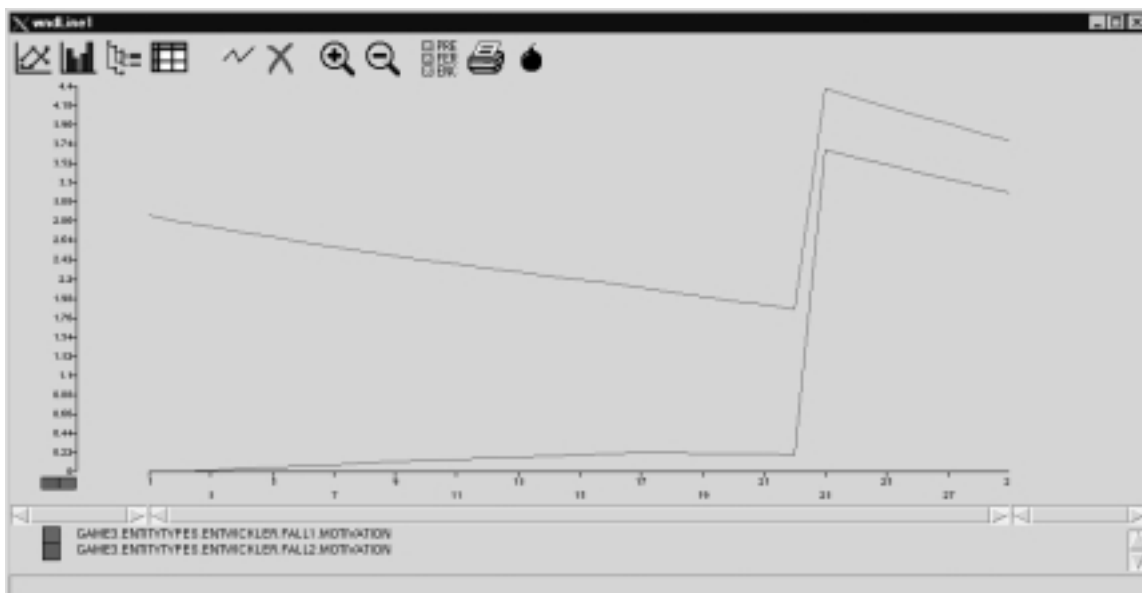


Abbildung 8: Testprotokoll Teamschulung (Motivation)

Fall 2

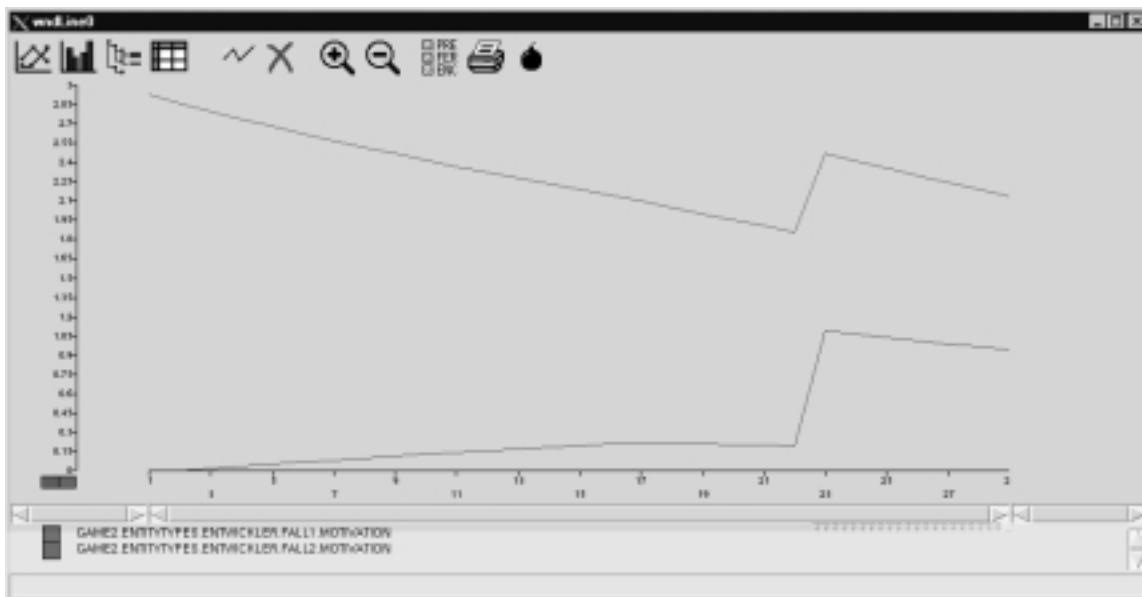


Abbildung 9: Testprotokoll Mittagessen (Motivation)

Die Validierung liefert die gewünschten Ergebnisse. Besonders deutlich ist die Auswirkung des Motivationsfaktors zu erkennen. So steigt die Motivation am Tag der Motivationsmaßnahme z.Bsp. im Fall 1 des durchschnittlich motivierten Mitarbeiters um etwa die volle Punktzahl (ca. 3.4) während die Motivation des hoch motivierten Entwicklers nur um ca. 2.6 steigt.

5.3 Die Art der Tätigkeit beeinflusst die Motivation

Wie bereits in der Modellierung beschrieben wurde, beeinflusst die Art der Tätigkeit in diesem Modell die Motivation (Kap. 4.2). Durch eine Tätigkeit in der ein Entwickler eine hohe Erfahrung besitzt wird die Motivation gesteigert, während eine geringe Erfahrung die Motivation senkt. Um diesen Effekt zu zeigen, werden zwei Mitarbeiter für ein Projekt eingestellt und bearbeiten über einen Zeitraum von 40 Tagen die Spezifikation. Beide starten mit einer durchschnittlichen Motivation. Der erste Mitarbeiter besitzt eine hohe Erfahrung in der Tätigkeit "Spezifizieren" während der andere Mitarbeiter keine Erfahrung beim Spezifizieren besitzt. Am Ende des Zeitraumes sollte sich die Motivation des ersten Entwicklers erhöht haben und er sollte hochmotiviert sein. Der zweite Entwickler sollte am Ende eine signifikant niedrigere Motivation besitzen als sein Kollege. Die Steigerung der Motivation bei einer Tätigkeit mit hoher Erfahrung sollte jedoch geringer ausfallen, als die Absenkung bei einer Tätigkeit mit niedriger Erfahrung, da es sich bei diesem Faktor um eine Hygienefunktion handelt (s.Kap. 2.2.1).

Testauswertung

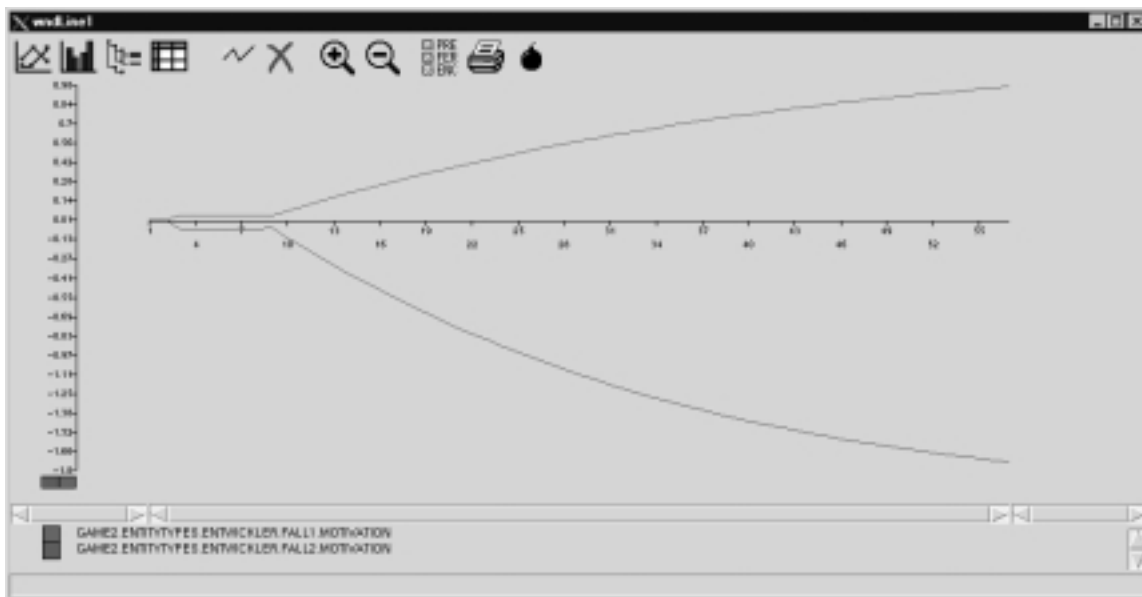


Abbildung 10: Testprotokoll Auswirkungen der Art der Tätigkeit auf die Motivation (Motivation)

Wie erwartet, wirkt sich die Tätigkeit signifikant auf die Motivation aus. Ab dem Beginn der Tätigkeit steigt die Motivation bei Entwickler 1 stark an, während sie bei Entwickler 2 stark abfällt. Auch der Unterschied zwischen dem Ausmaß der Steigerung und dem der Senkung der Motivation wird klar deutlich.

5.4 Veränderliche Erfahrungen

Anhand dieses Tests werden die neu eingefügten veränderlichen Erfahrungen überprüft. Übt ein Mitarbeiter eine Tätigkeit über einen gewissen Zeitraum aus, so erhöht sich auch seine Erfahrung in diesem Bereich. Die Auswirkungen einer Erhöhung der Erfahrung sind eine höhere Produktivität und eine bessere Motivation. Um dies zu zeigen, werden zwei Testläufe durchgeführt. In beiden Fällen startet der Mitarbeiter mit einer durchschnittlichen Motivation.

Fall 1

Ein Entwickler besitzt eine geringe Erfahrung in einer bestimmten Tätigkeit, die 20 Tage von der nächst höheren Stufe entfernt ist. Er arbeitet 40 Tage an einem Dokument.

Fall 2

Ein Entwickler besitzt eine geringe Erfahrung in einer bestimmten Tätigkeit, die 50 Tage von der nächst höheren Stufe entfernt ist. Er arbeitet 40 Tage an einem Dokument.

In Fall 1 wird der Entwickler nach 20 Arbeitstagen einen Sprung in die nächst höhere Erfahrungsklasse vornehmen. Dadurch steigt seine Produktivität und seine Motivation. In Fall 2 wird dies nicht der Fall sein. Der Entwickler aus Fall 1 hat also am Ende des Zeitraumes eine höhere Motivation und hat mehr Arbeit erledigt.

Testauswertung

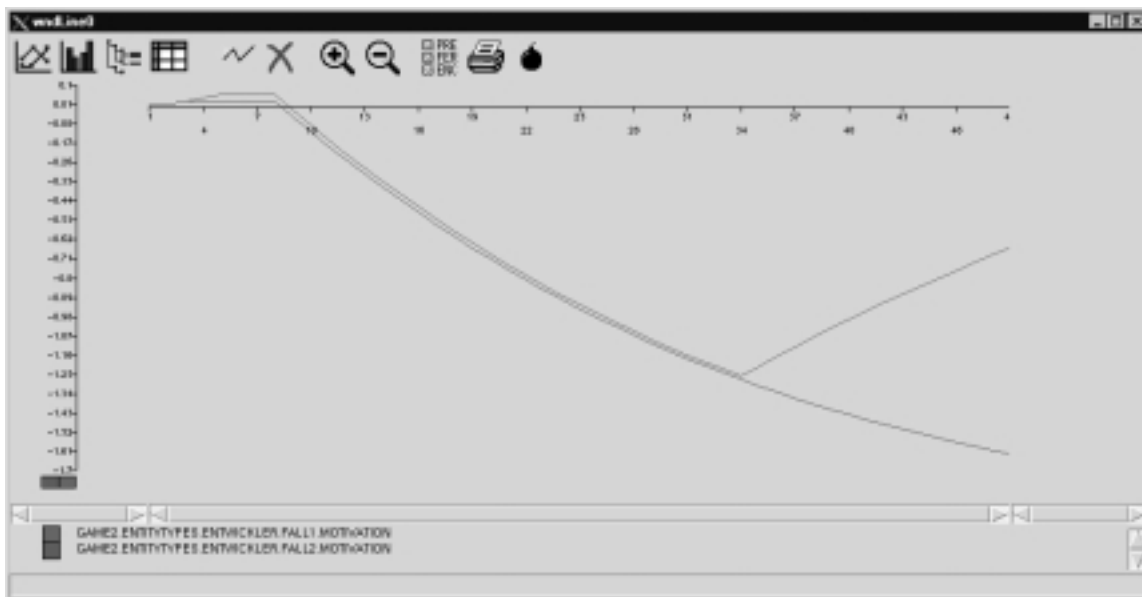


Abbildung 11: Testprotokoll veränderliche Erfahrungen (Motivation)

Man kann sehr gut erkennen, daß bei Entwickler 1 die Motivation anfangs, beeinflusst durch eine geringe Erfahrung, sinkt. Ab dem Zeitpunkt, an dem die Erfahrung auf die Stufe “MITTEL” wechselt, steigt die Motivation an und erholt sich. Bei Entwickler 2 findet der Aufstieg in die nächste Erfahrungsklasse nicht statt und so sinkt die Motivation weiter.

5.5 Überstunden

Im neu erstellten Modell ist es nun möglich, Überstunden anzuordnen. Überstunden bewirken eine Steigerung der Arbeitszeit um 25%. Dadurch hat der Mitarbeiter eine um 25% höhere Arbeitsleistung pro Tag. Werden für einen Mitarbeiter über einen Zeitraum von mehr als 10 Tagen Überstunden angeordnet, so reagiert er darauf mit einer Motivations-senkung. Überstunden werden aber nicht sofort nach Ende der Überstundenzeit vergessen. Der Mitarbeiter benötigt pro Tag, an dem er Überstunden machen mußte, zwei Tage um diese zu vergessen. Er benötigt also die doppelte Zeit, um Überstunden gedanklich wieder abzubauen. Diese Effekte werden mit zwei Testfällen gezeigt:

Fall 1

Ein Mitarbeiter arbeitet 50 Tage lang mit Überstunden an einer Tätigkeit. Die Erfahrung des Mitarbeiters in der Tätigkeit ist “MITTEL”, um die Motivation nicht zu stark zu beeinflussen.

Fall 2

Ein Mitarbeiter arbeitet 15 Tage lang mit Überstunden an einer Tätigkeit. Die Erfahrung des Mitarbeiters in der Tätigkeit ist “MITTEL”, um die Motivation nicht zu stark zu beeinflussen. Nach dieser Zeit arbeitet er 35 Tage ohne Überstunden weiter.

Fall 3

Ein Mitarbeiter arbeitet 50 Tage lang ohne Überstunden an einer Tätigkeit. Die Erfahrung des Mitarbeiters in der Tätigkeit ist "MITTEL", um die Motivation nicht zu stark zu beeinflussen.

Während der ersten 10 Tage muß die Arbeitsleistung in Fall 1 und Fall 2 25% höher sein als die in Fall 3. Durch die Absenkung der Motivation ab Tag 10 in Fall 1, senkt sich auch die Arbeitsleistung, so daß die Überstunden am Ende des Zeitraumes fast keinen Vorteil mehr bringen. Im Fall 2 sollte die Motivation bis Tag 25 sinken. Bis zum Tag 15 haben sich auf dem Überstundenkonto des Mitarbeiters 30 Überstunden angesammelt. Da bei einem Wert über 20 die Motivation sinkt, benötigt er weitere 10 Tage um die überzähligen 10 Überstunden zu vergessen. Ab dem 25. Tag bleibt die Motivation konstant.

Testauswertung

In den Abbildungen wird Mitarbeiter 1 mit GAME1 bezeichnet, Mitarbeiter 2 mit GAME2 und Mitarbeiter 3 mit GAME3.

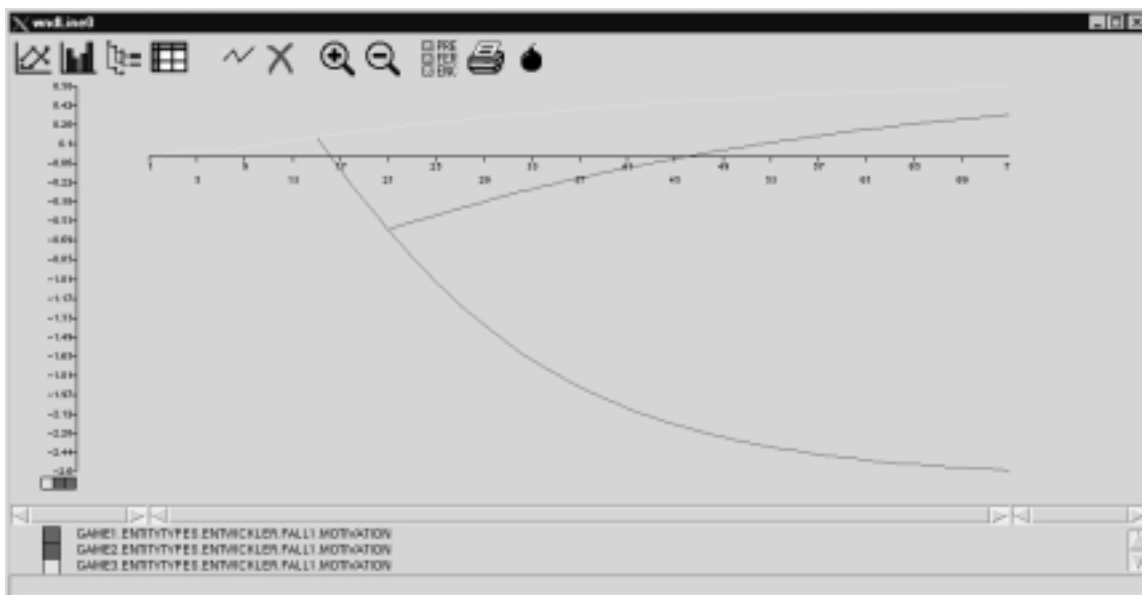
Motivation

Abbildung 12: Testprotokoll Überstunden (Motivation)

Die Motivation verhält sich wie erwartet. Mitarbeiter 1 besitzt aufgrund ständiger Überstunden ab den elften Tag der Tätigkeit (im Schaubild Tag 15 des Projektes) eine ständig schlechte Motivation. Mitarbeiter 2 besitzt eine sinkende Motivation bis zu dem Punkt, an dem die Überstunden vergessen sind. Von da an steigt die Motivation wieder. Mitarbeiter 3 besitzt ohne Überstunden eine ausgeglichene Motivation über das gesamte Projekt hinweg.

Verrichtete Arbeit

Zur Veranschaulichung wie produktiv die Mitarbeiter waren, wird in diesem Diagramm die Anzahl der Seiten der Spezifikation betrachtet.

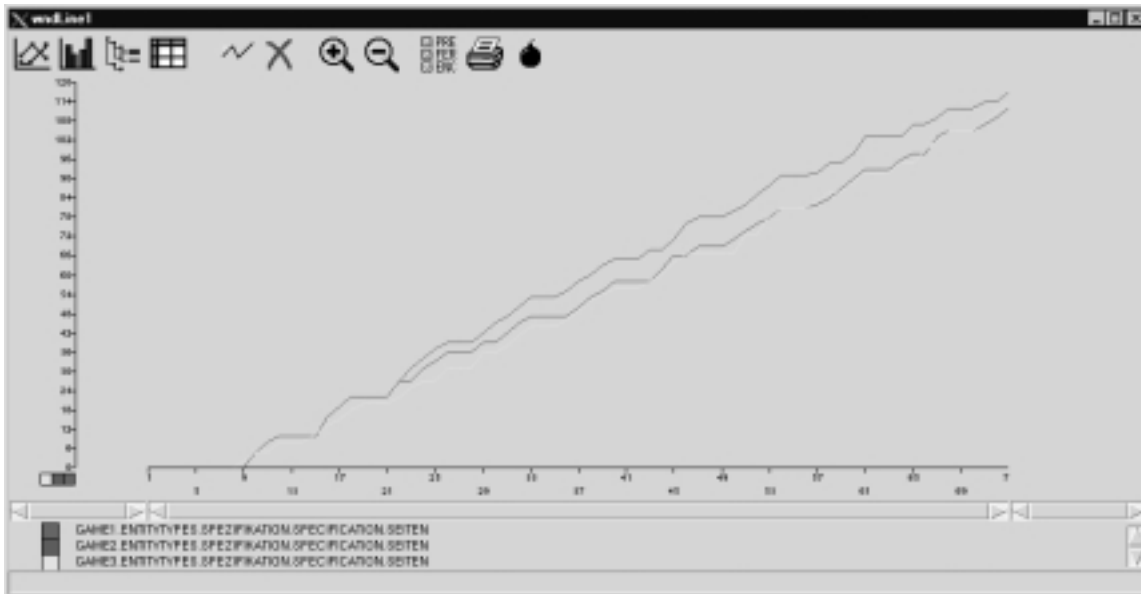


Abbildung 13: Testprotokoll Überstunden (Verrichtete Arbeit)

Hierbei gibt es interessante Effekte zu beobachten. Zuerst fällt auf, daß nach einer Zeit von 60 Arbeitstagen durch den Mitarbeiter, der ohne Überstunden arbeitete fast die gleiche Anzahl an Seiten erstellt wurde wie von dem Mitarbeiter, der über die gesamte Zeit hinweg mit 25% Überstunden gearbeitet hat.. Durch die Überstunden konnte zwar kurzfristig ein Vorsprung geschaffen werden, der aber mit der Zeit durch die schlechte Motivation zunichte gemacht wurde. Den schlechtesten Wert in dieser Kategorie lieferte Mitarbeiter 2, der den Vorsprung von Mitarbeiter 1 trotz besser werdender Motivation nicht mehr aufholen konnte. Dies belegt exakt die These, daß Überstunden nur bis zu einem bestimmten Zeitpunkt sinnvoll sind und dann keinen Vorteil gegenüber normaler Tätigkeit mehr bringen.

5.6 Die Motivation beeinflusst die Produktivität

Die Produktivität ist von der Motivation abhängig. Die Produktivität kann dadurch um bis zu 20% nach oben oder nach unten schwanken. Um diesen Effekt zu zeigen, werden drei Mitarbeiter eingestellt. Der eine Mitarbeiter bewegt sich dauerhaft auf einem hohen Motivationslevel. Dies wird durch eine Tätigkeit erreicht, in der er eine hohe Erfahrung besitzt. Der zweite Mitarbeiter wird durch seine Tätigkeit ständig auf einer durchschnittlichen Motivation gehalten. Ein dritter Mitarbeiter wird konstant auf einer niedrigen Motivationsstufe gehalten. Auch dies wird über die Tätigkeit beeinflusst, indem er nur eine geringe Erfahrung in dieser Tätigkeit besitzt.

Mitarbeiter 1 sollte am Ende des Testzeitraumes eine erheblich höhere Anzahl von AFP bearbeitet haben als Mitarbeiter 2

Testauswertung

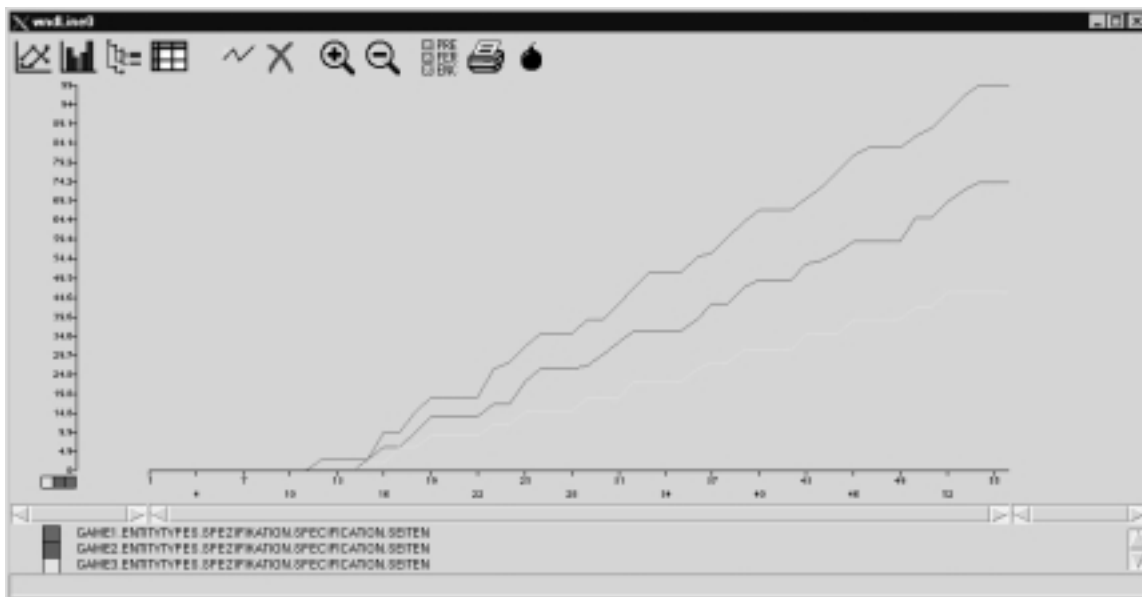


Abbildung 14: Testprotokoll Produktivität (Verrichtete Arbeit)

Auch hier werden die erwarteten Ergebnisse bestätigt. Die Unterschiede in der Produktivität durch die Motivation wird sehr plastisch ersichtlich.

5.7 Fehlzeiten sind von der Motivation abhängig

Im Bereich der Fehlzeiten gilt es zwei Aspekte zu zeigen. Erstens müssen die durchschnittlichen Fehlzeiten für das Modell stimmen. Zweitens muß gezeigt werden, daß die Fehlzeiten durch die Motivation beeinflusst werden. Da die Wahrscheinlichkeiten für Fehlzeiten sehr gering sind, kann hier nicht mit einem einzigen Projektdurchlauf validiert werden. Die Obergrenze für Projekte, für die das QS-Modell quantifiziert wurde, liegt bei 1000 AFP. Selbst bei einem solchen Projekt treten die Fehlzeiten nicht in dem Maße auf um sie *signifikant* unterscheiden zu können. Der Unterschied wäre zwar spürbar, aber es gäbe keine so hohe Anzahl von Fällen, um eine Wahrscheinlichkeit auch beweisen zu können. Aus diesem Grund wird ein Projekt mit relativ langer Dauer (ca. 2 Jahre) erstellt und dies mehrmals durchgeführt. Daraus müßte sich eine hohe Zahl von Fehlzeitfällen ergeben, so daß die feinen Unterschiede bewiesen werden können.

Im Testfall werden drei Mitarbeiter eingestellt. Der eine Mitarbeiter wird auf einer hohen Motivation gehalten (Arbeit mit hoher Erfahrung, keine Überstunden). Der zweite Mitarbeiter hat über das gesamte Projekt hinweg eine durchschnittliche Motivation. Der dritte Mitarbeiter wird auf einer sehr niedrigen Motivation gehalten (keine Erfahrung in der Tätigkeit, Überstunden).

Die Fehlzeiten in Fall 1 müssen niedriger sein als in Fall 2, diese wiederum niedriger als in Fall 3. Alle Fehlzeiten gemittelt müssen den durchschnittlichen Fehlzeiten entsprechen. Da durch eine niedrige Motivation kurze Fehlzeiten erhöht werden, sollte der Anteil der kurzen Fehlzeiten in Fall 3 höher als in Fall 1 sein.

Testauswertung

Bei der Testauswertung dieses Testfalles wurde eine Verhaltensweise des SESAM-Basis-systems entdeckt, die im voraus nicht bekannt war. So liefert die Zufallsfunktion keines-falls zufällige Werte, sondern bei verschiedenen Durchläufen der gleichen Vorgehensweise identische Werte. Verschiedene Zufallswerte treten nur bei verschiedenen Prozeßabläufen auf. Es hätte also für jeden Durchlauf des Testfalles eine neue Ablaufdatei erstellt werden müssen. Dieser Aufwand war im Rahmen dieser Arbeit nicht aufzubringen. Die Auswertung eines einzigen Durchlaufes ergab folgende Werte, die aber nur bedingt als repräsentativ angesehen werden können.:

Entwickler	Arbeitstage	Fehltage insgesamt	Anzahl Krankmel-dungen	Fehlzeiten 1 Tag	Fehlzeiten 2 Tage	Fehlzeiten 5 Tage
Fall 1	308 (100%)	14 (4.5%)	7 (100%)	3 (42%)	3 (42%)	1 (16%)
Fall 2	308 (100%)	19 (6.2%)	8 (100%)	3 (37%)	3 (37%)	2 (26%)
Fall 3	308 (100%)	23 (7.5%)	13 (100%)	6 (46%)	6 (46%)	1 (8%)

Tabelle 12: Testprotokoll Fehlzeiten

Die Zahlen weichen aufgrund der geringen Datenbasis in gewissem Maße von den Vorga-bewerten ab, lassen die quantifizierten Effekte aber erkennen. Aus dieser Erkenntnis folgt, daß die Validierung dieses Aspektes zu einem späteren Zeitpunkt wiederholt werden muß, indem eine Vielzahl unterschiedlicher Ablaufdateien bereitgestellt wird, um eine ausrei-chend häufige Ausführung des Prozesses mit verschiedenen Zufallswerten zu ermögli-chen.

5.8 Kündigung durch zufällige Ereignisse

Ereignisse, die außerhalb des Projektes und damit außerhalb des Modells stattfinden kön-nen dazu führen, daß Mitarbeiter kündigen. Da auch hier wiederum die Wahrscheinlich-keiten sehr gering sind, wird auf einen ähnlichen Trick zurückgegriffen wie im vorherigen Fall. Es wird ein relativ langes Projekt erstellt, das mehrmals durchgeführt wird. Werden alle Fälle zusammen betrachtet, muß sich die Wahrscheinlichkeit zeigen lassen. Es werden für das Projekt mehrere Mitarbeiter eingestellt, um den Fall, daß eine zufällige Kündigung eintritt, wahrscheinlicher zu machen. Die Mitarbeiter werden während des gesamten Pro-jektes auf einer konstant mittleren Motivationsstufe gehalten, um Auswirkungen der Moti-vation auf die Kündigung auszuschließen.

Testauswertung

Auch in diesem Testfall wirkten sich die Probleme der deterministischen Zufallsfunktion aus. Durch einen glücklichen Zufall wurde aber in dem einen Testlauf, der durchgeführt werden konnte, einmal eine zufällige Kündigung ausgelöst. So konnte zwar verifiziert werden, daß eine zufällige Kündigung ordnungsgemäß abläuft, die Wahrscheinlichkeit jedoch konnte nicht validiert werden, da hierfür eine Vielzahl von unterschiedlichen Durchläufen nötig gewesen wäre.

5.9 Schlechte Motivation führt zu Kündigung

Eine konstant schlechte Motivation führt einen Mitarbeiter dazu, zu kündigen. In etwa 50% der Fälle sucht er vorher seinen Projektleiter auf, um ihn auf die schlechte Motivation aufmerksam zu machen und vor einer Kündigung zu warnen.

In diesem Testfall wird die Motivation eines Mitarbeiters ständig gesenkt. Dies wird durch eine Tätigkeit, in der der Mitarbeiter keine Erfahrung hat und durch ständige Überstunden erreicht. Nach einem bestimmten Zeitraum kündigt der Mitarbeiter und verläßt dann das Projekt. In etwa 50% der Fälle sollte er zuvor seine Projektleiter zu einem Gespräch auffordern. Um die Wahrscheinlichkeit des Gesprächs testen zu können, wird der Versuch wiederum mehrere Male durchgeführt.

Testauswertung

Auch hier gilt das oben angesprochene Problem mit der deterministischen Zufallsfunktion. Die Motivation des Mitarbeiters wurde soweit gesenkt, daß er eine Kündigung aufgrund schlechter Motivation einreichte. Jedoch verlangte er nach keinem Mitarbeitergespräch, um den Projektleiter darauf aufmerksam zu machen. So konnte zwar der Prozeß der Kündigung aufgrund schlechter Motivation verifiziert werden, eine Verifizierung und Validierung der Wahrscheinlichkeit konnte aber nicht durchgeführt werden, da eine ausreichende Anzahl von Testdurchläufen nicht durchgeführt werden konnte.

Kapitel 6:

Bewertung und Ausblick

In diesem Kapitel möchte ich einmal kritisch auf meine Diplomarbeit zurückblicken und kurz darüber reflektieren, welche Ergebnisse wie zustande gekommen sind und welche neuen Ideen sich aus dieser Arbeit ergeben. Im Gegensatz zu allen vorherigen Kapiteln die wertfrei gehalten wurden, möchte ich hier meine persönliche Meinung einbringen. Der Wechsel im Erzählstil soll dies dem Leser ständig in Erinnerung behalten.

6.1 Ein persönlicher Rückblick

Als ich mit dieser Arbeit begonnen habe stand über lange Zeit die Frage im Raum, ob diese Arbeit vernünftige Ergebnisse liefern könne. Die Literatur in diesem Bereich war sehr dünn gesät und es wurde bereits sehr früh klar, daß sich das Thema nicht nur im Bereich des Software Engineerings bewegt, sondern auch starke Berührungspunkte mit anderen Wissenschaften wie der Psychologie und der Betriebswirtschaftslehre hat. Doch dies erwies sich keinesfalls als negativ, sondern lieferte neue Sichtweisen auf ein bisher recht unerforschtes Gebiet. Viele Stunden in Bibliotheken anderer Fakultäten brachten schließlich eine Vielzahl von sehr interessanten Aspekten zu Tage, die im Rahmen dieser Arbeit niemals alle hätten realisiert werden können. So stand ich letztendlich sogar vor der “Qual” einer Auswahl, in der ich Aspekte aussortieren konnte (oder mußte), die den Konzepten, die hinter SESAM und dem QS-Modell stehen, am wenigsten entsprachen. Vielleicht wurde dem einen oder anderen Leser bei der Lektüre des zweiten Kapitels bewußt, wie vielschichtig und kompliziert menschliche Verhaltensaspekte sind und welche Verflechtungen sich daraus ergeben können. Selbst in einem doch oft so sehr durchgeplanten

Bereich wie einem Softwareprojekt treten diese Effekte auf, wahrscheinlich sogar häufiger als man denkt.

Die Modellierung versetzte mich über die ganze Zeit hinweg in Gewissenskonflikte, galt es doch ständig den Spagat zwischen der Realitätsnähe, der Umsetzbarkeit und dem Lerneffekt zu meistern. Man versucht immer den Punkt zu finden, an dem diese drei Aspekte in gleichem Maße berücksichtigt werden. Hunderprozentig gelingt es einem jedoch nie. Ich mußte immer in dem einen oder anderen Punkt Abstriche machen. Entsprechend oft wurde das Zwischendokument "Umsetzung der Zusammenhänge von Softwareprojekten in SESAM", das in etwa dem Kapitel 3 in diesem Bericht entspricht, umgestellt, neu geschrieben und gereviewt. Diese Tatsache fand sich dann auch im Projektplan wieder. Nach Abschluß dieses Dokumentes waren sämtliche Zeitpuffer, die zu Beginn der Arbeit großzügig in den Plan eingebaut wurden verbraucht, und darüber hinaus sind noch zwei zusätzliche Wochen vergangen. Die Zeit war es aber wert. Die Ergebnisse dieser Phase sind meiner Ansicht sehr interessant geworden.

Die Quantifizierung stellt mich vor das Problem, die bereits weit zurückliegenden Inhalte des Vordiploms, insbesondere der Höheren Mathematik, wieder ins Gedächtnis zurückzurufen. Ich stand vor der fast schon paradoxen Aufgabe, menschliches Verhalten in mathematische Formeln pressen zu müssen. Eine Aufgabe, die nur durch starke Verkürzungen und Kompromisse möglich war. Da außerdem zu diesem Thema noch weniger Literatur zur Verfügung stand als schon zuvor, mußte ich meine Quantifizierung teilweise auf hypothetische Beine stellen. Jedoch sollte ja am Ende der Arbeit versucht werden, diese Hypothesen durch eine Validierung bzw. durch Plausibilitätsprüfungen zu stärken.

Während der Implementierung lernte ich die Schwierigkeiten eines regelbasierten Systems kennen. Wer sich einmal durch ein solches System gewühlt hat und nach mehreren Stunden Modultest immer noch nicht verstehen kann warum eine bestimmte Regel nun feuert oder nicht, der kann diese Umstände sicherlich nachvollziehen. Dadurch, daß sich die Aspekte, die ich implementiert habe fast nicht modularisieren ließen bzw. nicht in der Weise modularisieren ließen, in der das bisherige QS-Modell modularisiert ist, mußte ich fast über den gesamten Zeitraum hinweg auf der gesamten Regelmenge arbeiten. Dies erschwerte nicht nur das Verständnis der aktuellen Situationen, sondern führte auch zu Kompilierzeiten, die teilweise im Bereich einer Viertelstunde und darüber lagen. Letztendlich aber war es dann doch geschafft und alle Aspekte, die modelliert wurden sind auch implementiert.

Die Validierung bereitete mir einige Probleme. Einerseits konnte so gut wie keine Literatur gefunden werden, anhand der die implementierten Aspekte hätten validiert werden können, andererseits rückte der Endtermin der Arbeit bereits in drohende Nähe. Nachdem ich zu Beginn der Validierungsphase eine Vielzahl von Testfällen erarbeitet hatte, um alle möglichen und unmöglichen Kombinationen und Prozesse testen zu können, mußte ich dann im Laufe der Zeit feststellen, daß im Rahmen dieser Arbeit nur einige ausgewählte

Fälle bearbeitet werden konnten. Ich habe die wichtigsten Fälle herausgegriffen und diese validiert.

Abschließend kann ich sagen, daß ich mit meiner Arbeit sehr zufrieden bin. Ich habe ein Vielzahl an interessanter Aspekte und Zusammenhänge gefunden und die für alle Seiten interessantesten davon modelliert und in das QS-Modell integriert. Auch die Quantifizierung erscheint plausibel und liefert die erwarteten Resultate. Ich persönlich bin bereits sehr gespannt, ob das erweiterte QS-Modell sich irgendwann einmal in der Praxis beweisen muß und wie es dabei abschneidet.

6.2 Was kommt danach?

Wie bereits im Rückblick erwähnt, mußten einige der identifizierten Aspekte, die sicherlich interessant sind, aus der Arbeit herausgenommen werden, da sie den Zeitrahmen gesprengt hätten. Ein Aspekt, den ich mir noch in einem Projektmanagement-Simulator vorstellen könnte, wäre die Personalführung. Hier könnte versucht werden, etwas genauer auf die Beziehung zwischen dem Projektleiter und den Mitarbeitern einzugehen. Dies ist jedoch eine sehr komplizierte und umfassende Aufgabe, die sehr viel Hintergrundinformation und -forschung benötigt. Ein weiterer interessanter Aspekt sind die Anforderungsänderungen. Im Verlauf der Arbeit stellte sich heraus, daß durch die Hinzunahme von Anforderungsänderungen das gesamte System neu quantifiziert werden müßte. Dies würde eine komplette Neuvalidierung nach sich ziehen; ein erheblicher Aufwand. In Gesprächen mit Projektleitern aus der Praxis hört man aber erschreckende Zahlen bezüglich Anforderungsänderungen. So ist es beispielsweise bei der IBM nicht unüblich, daß bis zu 50% und mehr der Anforderungen sich im Laufe eines Projektes ändern. In diesem Bereich sind sicherlich noch interessante Aufgaben zu vergeben.

Die oben genannten Erweiterungspunkte lassen sich sicherlich unter dem Sprichwort “nice to have”, also Punkte die nicht unbedingt eingearbeitet werden müssen, einordnen. Zwei Punkte liegen mir jedoch am Herzen, die ich in die Kategorie “must have”, also unbedingt durchzuführen, einordnen würde. Hier ist an erster Stelle eine ausführliche Validierung der neuen Aspekte zu nennen. Wie bereits erwähnt, sind bereits zahlreiche Testfälle definiert, die alle neuen Aspekte prüfen und anhand derer dann Plausibilitätsprüfungen durchgeführt werden sollten. Aufgrund der vielen Hypothesen in der Arbeit gibt es eine Vielzahl von Effekten, die genauestens überprüft werden müssen.

Der zweite “must have”-Punkt ist ein Konzept, das sicherstellt, daß die Möglichkeiten, die durch das neue Modell nun zur Verfügung stehen, richtig genutzt werden. Es muß also ein didaktischen Konzept erarbeitet werden, das sowohl das alte Modell als Einstiegsmodell, als auch das erweiterte Modell als Modell für Fortgeschrittene in eine Ausbildung für Projektleiter einbettet. Die beiden letzten Punkte sind aber bereits heute in Angriff genommen, so daß ich keine Bedenken habe, daß sie zu kurz kommen.

Kapitel 7:

Literaturverzeichnis

[AOK98]

Wissenschaftliches Institut der AOK, **Krankheitsbedingte Fehlzeiten in der deutschen Wirtschaft - Branchenreport 1997**, , Bonn, Juli 1998

[Barto97]

Bartol K.M., Martin D.C., **Managing the Consequences of DP Turnover: A Human Resource Planning Perspective**, in Thayer R. (Ed.), Software Engineering Project Management ,2nd Ed., IEEE Computer Society Press, 1997

[Boehm86]

Boehm Barry W., **Wirtschaftliche Software-Produktion**, Wiesbaden, 1986

[Boddi99]

Boddie J., **What are we going to do about Alice?**, in: Cutter IT Journal, Vol.12, Nr.1, Jan. 1999

[Brook74]

Brooks F.P., **The Mythical Man Month**, Addison Wesley, 1974

[Chand71]

Chandor A., **Choosing and keeping Computer staff**, 1971

[Couga78]

Cougar J.D., Zawacky R.A., **What motivates a DP Professional?**, Datamation September 1978

[Davis97]

Davis A., **Trial by firing: Saga of a Rookie Manager**, in Thayer R. (Ed.), Software Engineering Project Management ,2nd Ed., IEEE Computer Society Press, 1997

[deMar91]

deMarco T., **Wien wartet auf dich (engl. Orig.: Peopleware: Productive Projects and Teams)**, Hanser Verlag, München, 1991

[Drapp98]

Drappa A., Ludewig J., **Quantitative Modeling for the Interactive Simulation of Software Projects**, Proceedings of the Software Process Simulation Modeling Workshop Pro-Sim 98, Silver Falls, OR, 1998

[Drapp00]

Drappa A., **Quantitative Modellierung von Softwareprojekten**, Shaker, Aachen, 2000

[Elzer89]

Elzer P.F, **Management von Softwareprojekten**, Informatik-Spektrum (1989) 12

[Hentz95]

Hentze, J. **Personalwirtschaftslehre 2**, 6. Aufl., Bern, Stuttgart u.a., 1995

[Herzb59]

Herzberg F., Mausner B., Snyderman B.B., **The Motivation to Work**, New York 1959

[Jones86]

Jones C., **Applied Software Measurement**, 1996

[Jones96]

Jones C., **Conflict and Litigation between Software Clients and Developers**, 1996

[Ludew98]

Ludewig J, **Der Software-Projekt-Manager. Domteur, Coach oder Mädchen für alles?**, Schweizerische Technische Zeitschrift 95.Jhg, Nr. 9/1998, S. 26-30

[Ludew99]

Ludewig J., **Skript zur Vorlesung Software Engineering**, Universität Stuttgart, 1999

[Mandl99]

Mandl-Striegnitz P., Lichter H., **Defizite im Software-Projektmanagement - Erfahrungen aus eine industriellen Studie**, Informatik/Informatique, 5/99, 1999

[Mandl99a]

Mandl-Striegnitz P., **Erweiterung eines SESAM-Projektmodells um Verhaltensaspekte**, Aufgabenstellung zur Diplomarbeit, 1999

[Mandl00]

Mandl-Striegnitz P., Lichter H., **Untersuchung eines neuen Ansatz zur Projektmanagement-Ausbildung**, in Software-Metriken - Entwicklungen, Werkzeuge und Anwendungs-verfahren, Dumke, R. Lehner, F. (Hrsg.), Deutscher Universitäts-Verlag, Wiesbaden, 2000

[Maslo54]

Maslow A.A., **Motivation and Personality**, Harper & Row, New York, 1954

[McDon99]

McDonald M.P., **Building and Sustaining High-Performing Teams**, in: Cutter IT Journal, Vol.12, Nr.1, Jan. 1999

[Metzg96]

Metzger P., Boddie J., **Managing an Programming Project - Processes and People**, 3rd Ed., Prentivce Hall, 1996

[Powel97]

Powell G.N., Posner B.Z., **Excitement And Commitment: Keys To Project Success**, in: Thayer R. (Ed.), Software Engineering Project Management ,2nd Ed., IEEE Computer Society Press, 1997

[Rothm99]

Rothman J., **What to Do When The Right Person Doesn't Come Along**, in: Cutter IT Journal, Vol.12, Nr.1, Jan. 1999

[Staeh87]

Staehle, W.H., Sydow, J., **Führungsstiltheorien**, in Handwörterbuch der Führung, hrsg. v. Kieser, A. u.a., Stuttgart 1987

[Tanne58]

Tannenbaum R., Schmidt W.H., **How to Choose a Leadership Pattern**, in: HBR, Vol.36 (1958)

[Thaye97]

Thayer R. (Ed.), **Software Engineering Project Management** ,2nd Ed., IEEE Computer Society Press, 1997

[Weltz92]

Weltz F., Ortmann R.G., **Das Softwareprojekt**, Campus Verlag, 1992

[Whita97]

Whitaker K., **Motivating and Keeping Software Developers**, in: Thayer R. (Ed.), Software Engineering Project Management ,2nd Ed., IEEE Computer Society Press, 1997

Ich versichere, dass ich diese Arbeit selbständig verfasst und nur die angegebenen Hilfsmittel verwendet habe.

Dettenhausen, den 15. Mai 2000

Axel Dudler
